SEID-Prosjektet

Leveranse oppgave 3

SEID-SDO –
Dataobjekt for langtidslagring og utveksling av elektroniske signaturer

Versjon 1.01

Status: Godkjent

Dato: 01.06.2012

HISTORIKK

Dato	Versjon	Utført av	Kommentar
01.06.05	1.0	PK	Dokumentet godkjent av SEID-prosjektets styringsgruppe.
01.06.12	1.01	PK	Fjernet attributet ”ObjectRefrence” i Bilag A B.1. Oppdatert kontaktpunkt for ansvarlig dokumentforvalter i kap. 5.1.

INNHOLDSFORTEGNELSE

1 Begreper og forkortelser

Alle begrepene i tabellen nedenfor er definert med stor forbokstav. For å lette lesbarheten av dokumentet er stor forbokstav tilsvarende benyttet når de samme begrepene er anvendt i øvrige deler av dokumentet.

Begrep	**Forklaring**
Avansert Elektronisk Signatur	Elektronisk signatur som tilfredsstiller nærmere spesifiserte krav slik de er definert i Lov om elektronisk signatur [16].
BankID_SDO	Standardisert dataobjekt [12] fra Bankenes Standardiseringskontor til bruk innenfor BankID Samarbeidet for oppbevaring av Elektroniske Signaturer over tid.
Basis Signatur	Elektronisk Signatur representert ved et dataobjekt som er formatert i henhold til CMS [7] eller XMLDSIG [8] og som oppfyller eventuelle minimumskrav som stilles til objektet ved lagring i et SDO Signaturelement som en del av SEID-SDO. Begrepet Basis Signatur tilsvarer det ETSI kaller for BES eller EPES i sine signaturstandarder (se Bilag C).
Datavalideringssertifikat	Et Datavalideringssertifikat fungerer som et bevis fra 3.part for at en Elektronisk Signatur, i dette dokumentet en Basis Signatur, er verifisert å være gyldig på det tidspunktet som er angitt i sertifikatet. Se også Datavalideringstjeneste.
Datavalideringstjeneste	Tjeneste [13] levert av en Tillitstjenesteleverandør og som tilbyr 3.parts validering av en Elektronisk Signatur. Som et resultat av valideringen vil Tillitstjenesteleverandøren utstede et Datavalideringssertifikatet. For en aktør som har behov for å validere den Elektroniske Signaturen i ettertid vil det da kunne være nok å verifisere gyldigheten av Datavalideringssertifikatet og således stole på valideringen som Datavalideringstjenesten allerede har gjort fremfor å foreta en fullstendig validering av den Elektroniske Signaturen selv. Se også Datavalideringssertifikat.
Elektronisk Signatur	I dette dokumentet omfatter begrepet kun Avanserte Elektroniske Signaturer som er laget ved hjelp av ”offentlig nøkkel” kryptografi (*eng: public key cryptography*) og hvor signaturen er entydig knyttet til Undertegneren, og således identifiserer denne, gjennom et utstedt Signatursertifikat.
Kvalifisert Signatur	En Avansert Elektronisk Signatur som er basert på et såkalt kvalifisert sertifikat og fremstilt av et godkjent sikkert signaturfremstillingssystem (se for øvrig lov om elektronisk signatur [16]).
SDO Signaturelement	Subsett av et SEID-SDO objekt som inneholder èn Basis Signatur og tilhørende Valideringsdata. Tre ulike typer Signaturelementer støttes av SEID-SDO (se kap. 6.4).
SDO objekt	Subsett av et SEID-SDO objekt som inneholder ett eller flere SDO Signaturelementer samt en ytre datastruktur med et sett av data som er felles for alle Signaturelementene.
SEID-prosjektet	Prosjektet som har produsert dette dokumentet. Prosjektets fullstendige navn er “Samarbeidsprosjekt om eID og eSignatur”.
SEID-SDO	Dataobjekt utarbeidet av SEID-prosjektet for oppbevaring av

Begrep	Forklaring
	Elektroniske Signaturer over tid og med format og innhold som definert i dette dokumentet. Et SEID-SDO objekt kan inneholde ett eller flere SDO objekter.
Sertifikat	Et sertifikat er en form for elektronisk identitetsbevis utstedt av en Sertifikatutsteder. Sertifikater kan bl.a. anvendes som elektronisk legitimasjon eller for å validere en Elektronisk Signatur.
Sertifikatinnehaver	Den kunden (person/virksomhet) sertifikatet er utstedt til i henhold til sertifikatpolicy og som er innehaver av den private nøkkelen.
Sertifikatutsteder	En Sertifikatutsteder som omtalt i dette dokumentet vil være: • en juridisk person, dvs. et rettssubjekt som ikke er en fysisk person, i dette tilfelle en organisasjon. • ansvarlig for sertifikatutstedelsen, dvs, ansvarlig for implementeringen av sertifikatpolicy (selv om den operative utførelsen kan foretas av en annen aktør). • avtalepart for Sertifikatinnehaver. • erstatningsmessig ansvarlig i henhold til gjeldende erstatningsbestemmelser i relevante nasjonale lover, forskrifter samt i sertifikatpolicy for sertifikatene som utstedes.
Sertifikatvalideringsdata	Fellesbetegnelse på sertifikater og tilhørende sertifikatstatusinformasjon (dvs. CRLer og/eller OCSP [5] responser) som er nødvendig for å validere en Elektronisk Signatur.
Signaturattributt	Informasjonselement som sammen med Undertegnerens Dokument utgjør input til en elektronisk signeringsprosess.
Signatursertifikat	Et sertifikat som inneholder offentlig nøkkel som kan benyttes for å verifisere en Elektronisk Signatur.
Signaturpolicy	Et sett av regler for generering og validering av en Elektronisk Signatur og som definerer de tekniske og prosedyremessige krav for signerings- og valideringsprosessen som skal til for å imøtekomme et konkret forretningsbehov.
Tidsstempel	Signert tidsangivelse [14] fra en Tidsstemplingsautoritet (Tillitstjenesteleverandør) og som attesterer at nærmere angitte data, f.eks. en Basis Signatur, eksisterte på det angitte tidspunkt.
Tillitstjenesteleverandør	Leverandør av tjeneste som har som formål å bidra til å bygge tillitsrelasjoner mellom aktører i en samhandling.
Undertegner	Den person/aktør som har satt igang en elektronisk signeringsprosess og gjennom dette generert en Elektronisk Signatur. I dette dokumentet vil Undertegneren være en Sertifikatinnehaver.
Undertegnerens Dokument	Betegnelse på det dataobjekt som skal elektronisk signeres av Undertegner.
Valideringsdata	Fellesbetegnelse på all den tilleggsinformasjon som i en gitt kontekst må være tilgjengelig for aktører som har behov for å validere en Basis Signatur over tid. Sertifikater og informasjon som angir status for disse (eks. CRLer, OCSP responser) er eksempler på slik tilleggsinformasjon. Tidsstempel og Datavalideringssertifikat er andre eksempler.
XML Skjema	(eng: XML schema) XML basert formalisme for å beskrive regler for syntaks, struktur og verdier for instanser av XML dokumenter.

Forkortelse	Forklaring
ASCII	American Standard Code for Information Interchange
ASN.1	Abstract Syntax Notation no.1
BES	Basic Electronic Signature
CAdES	CMS Advanced Electronic Signatures
CMS	Cryptographic Message Syntax
CRL	Certificate Revocation List
eID	Elektronisk ID
EPES	Explicit Policy-based Electronic Signature
ES-A	Electronic Signature with Archiving validation data
ES-C	Electronic Signature with Complete validation reference data
ES-T	Electronic Signature with Time Stamp
ES-X	Electronic Signature with eXtended validation data
ETSI	European Telecommunications Standards Institute
IETF	Internet Engineering Task Force
ISO	International Organization for Standarization
OCSP	On-line Certificate Status Protocol
OID	Object IDentifier
PDF	Portable Document Format
PKI	Public Key Infrastructure
PKCS	Public Key Cryptograhic Standards
RFC	Request For Comments
SDO	Signert DataObjekt
SEID	Samarbeidsprosjekt om eID og eSignatur
SEID-SDO	SEID Signert DataObjekt
SHA	Secure Hashing Algorithm
TS	Technical Standard
WAP	Wireless Application Protocol
XAdES	XML Advanced Electronic Signatures
XML	eXtensible Markup Language
XMLDSIG	XML Digital SIGnature

2 Referanser

[1] PKI Forum, Strategi for en samfunnsinfrastruktur for elektronisk signatur og elektronisk ID i Norge, juni 2002, www.handel.no/pkiforum

[2] PKI Forum, Handlingsplan, Rapport fra "Midlertidig Prosjektgruppe" for oppfølging av PKI strategien, februar 2003, www.handel.no/pkiforum

[3] Nærings- og handelsdepartementet, "eNorge 2005", mai 2002, www.enorge.org

[4] IETF, RFC 3280, "Internet X.509 Public Key Infrastructure Certificate and CRL Profile", april 2002

[5] IETF, RFC 2560, Online Certificate Status Protocol (OCSP), juni 1999, www.ietf.org/rfc

[6] World Wide Web consortium, XML, Extensible Markup Language, april 2004, www.w3.org/TR/xml11

[7] IETF, RFC 3852, Cryptographic Message Syntax (CMS), juli 2004, www.ietf.org/rfc (erstatter RFC 3369)

[8] IETF, RFC 3275: XML-Signature Syntax and Processing, mars 2002, www.ietf.org/rfc. (Dokumentet er også publisert av World Wide Web consortium, www.w3.org/TR/xmldsig-core)

[9] ETSI TS 101 733, Electronic Signatures and Infrastructures (ESI); Electronic Signature Formats, versjon 1.5.1, desember 2003

[10] ETSI TS 101 903, XML Advanced Electronic Signatures (XAdES), versjon 1.2.2, april 2004

[11] IETF, RFC 3126: Electronic Signature Formats for long term electronic signatures, september 2001, www.ietf.org/rfc

[12] Bankenes Standardiseringskontor: BankID_ Signed Data Object (BankID_SDO Formatbeskrivelse, versjon 1.0, oktober 2003

[13] IETF, RFC 3029, Data Validation and Certification Server Protocols, februar 2001, www.ietf.org/rfc

[14] IETF, RFC 3161, Time-Stamp Protocol (TSP), august 2001, www.ietf.org/rfc

[15] WAP Forum, WML Script Crypto Library, version 20, juni 2001 www.openmobilealliance.org/tech/affiliates/wap/

[16] Lov 15. juni 2001 nr. 81 om Elektronisk Signatur

[17] IETF, RFC 3447, PKCS #1: RSA Cryptography Specifications Version 2.1, februar 2003, www.ietf.org/rfc

[18] ETSI TS 101 861, Time Stamping Profile, versjon 1.2.1, mars 2002

[19] ISO 8601, Data elements and interchange formats -- Information interchange -- Representation of dates and times, desember 2004

[20] Forskrift 1.juli 2004 nr.988 om elektronisk kommunikasjon med og i forvaltningen (eForvaltningsforskriften)

[21] IETF RFC 2315, PKCS #7: Cryptographic Message Syntax Version 1.5, mars 1998, www.ietf.org/rfc

[22] IETF, RFC 3369, Cryptographic Message Syntax (CMS), august 2002, www.ietf.org/rfc (erstattet av RFC 3852)

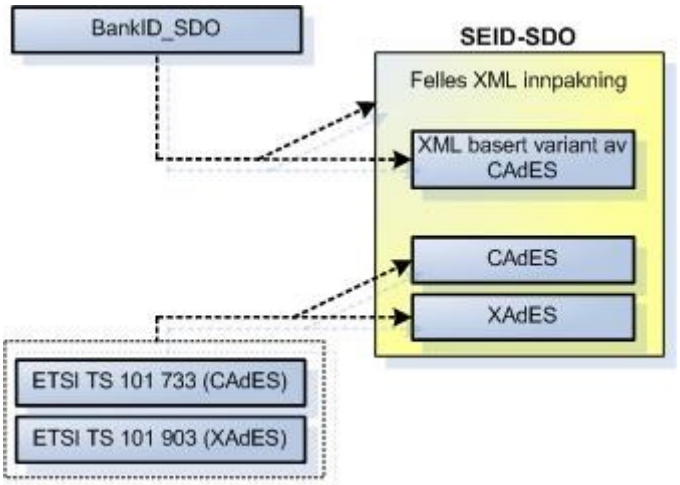
3 Sammendrag

I en nasjonal kontekst vil Elektroniske Signaturer oppstå gjennom ulike typer av samhandling mellom aktører og med ulike Tillitstjenesteleverandører (eks. tilbydere av valideringstjenester og Sertifikatutstedere) involvert. For å sikre teknisk åpenhet skal Signaturen med de nødvendig Valideringsdata kunne utveksles og valideres av vilkårlige aktører i ettertid.

Formålet med dette dokumentet er å spesifisere et dataobjekt (heretter kalt SEID-SDO) som skal kunne benyttes til både lagring og utveksling av Elektroniske Signaturer som skal kunne valideres over tid. Spesifikasjonen skal danne et best mulig grunnlag for harmonisering av løsninger fra ulike aktører i det norske markedet.

SEID-SDO er et påbygg til ETSI standardene ETSI TS 101 733 [9] og ETSI TS 101 903 [10]. SEID-SDO er samtidig en videreutvikling av BankID_SDO [12] som er et tilsvarende dataobjekt utviklet av banknæringen i Norge til bruk innenfor BankID Samarbeidet.

SEID-SDO er med andre ord en XML [6] struktur som støtter lagring av tre ulike typer signaturobjekter som vist i Figur 3-1. Kapittel 6.4 gir en mer detaljert beskrivelse av de ulike typene signaturobjekter.



Figur 3-1 – Introduksjon til SEID-SDO

4 Innledning

4.1 Bakgrunn

Nasjonalt PKI Forum la i juni 2002 frem en strategi for en samfunnsinfrastruktur for elektronisk signatur og elektronisk ID i Norge [1]. Strategien fikk bred tilslutning i en offentlig høring høsten 2002. I oppfølgingen av strategien utarbeidet PKI Forum en Handlingsplan [2] som er grunnlaget for etableringen av et samarbeidsprosjekt for eID og eSignatur, kalt SEID–prosjektet. Prosjektet hadde oppstart i november 2003 og er basert på en avtale mellom Nærings- og handelsdepartementet (NHD), Arbeids- og administrasjonsdepartementet (AAD) og 14 private virksomheter[1]. Prosjektet er blant annet forankret i Regjeringens IT politikk stadfestet i eNorge 2005 [3].

[1] NHD og AAD ble i løpet av 2004 slått sammen til Moderniseringsdepartementet (MOD).

4.2 Arbeidsgruppens mandat og medlemmer

Dette dokument inneholder SEID-prosjektets leveranse nummer tre med følgende mandat og oppgavedefinisjon.

Med utgangspunkt i aktuelle løsninger og løsningskrav fra markedsaktørene i prosjektet, samt relevante internasjonale standarder, skal det spesifiseres:

- *Et felles XML format (herunder XML Schema) som beskriver syntaks og semantikk for et signert objekt (informasjonscontainer). Informasjonscontaineren skal som minimum kunne benyttes til arkivering av elektroniske signaturer formatert ihht. CMS/PKCS#7 (RFC 3369).*
- *Den samme informasjonscontaineren skal kunne støtte arkivering av XML formaterte signaturer (XMLDSIG) som en opsjon. I tillegg til selve signaturdataene skal informasjonscontaineren kunne arkivere nødvendig tilleggsinformasjon (eks. tidsstempel) til bruk for bl.a. signaturvalidering ved en eventuell disputt om signaturens gyldighet i ettertid.*
- *En hensiktsmessig klassifisering av ulike signaturprofiler til bruk for ulike hovedtyper av signaturanvendelser/signaturpolicyer og dertil ulike behov for lagring av tilleggsinformasjon skal vurderes og beskrives.*
- *Til tross for at leveransen skal definere en spesialtilpasset XML struktur som skal støtte lagring av både PKCS#7/CMS formaterte signaturer og XML formaterte signaturer så vil de to allerede ETSI standardiserte signerte objektene i henholdsvis ETSI TS 101 733 og ETSI TS 101 903 anerkjennes som reelle alternativer.*

Arbeidsgruppen har bestått av:

- Pål Kristiansen, UniBridge AS (innleid prosjektleder og leder av gruppen)
- Rune Hagen, BankID Samarbeidet
- Bjørn Søland, BankID Samarbeidet
- Atle Dingsør, Buypass AS
- Lise Blix, DnB NOR ASA
- Per Myrseth, IBM Norge AS og DNV
- Kåre Langedrag, Posten Norge AS
- Tor Hjalmar Johannessen, Telenor ASA
- Per Christian Foss, Telenor ASA
- Ståle Gullbrekken, SpareBank 1 Gruppen AS
- Andreas Strand, Nordea Bank Norge ASA
- Håkon Liberg, IBM Norge AS

4.3 Formål, målgruppe, omfang og avgrensninger

Formål

Bruk av Elektroniske Signaturer knyttet til elektroniske tjenester og elektronisk saksbehandling vil kunne medføre krav til oppbevaring og evt. langtidslagring[2] av elektronisk signert materiale.

En av hovedutfordringene ved langtidslagring av elektronisk signert materiale er å ta vare på og, ved behov, utveksle all tilleggsinformasjon som er nødvendig for at signaturen skal kunne valideres i ettertid. For PKI-baserte Elektroniske Signaturer vil tilleggsinformasjon typisk omfatte Sertifikater og annen tilknyttet Valideringsdata (eks. CRLer [4] og OCSP [5] responser) som er nødvendig for å kunne bekrefte gyldigheten av Sertifikatene på det tidspunkt signaturen ble laget.

En annen utfordring ved langtidslagring er at enhver Elektronisk Signatur av natur har en tidsbegrenset kryptografisk levetid samt potensielt svakheter ved de hash-algoritmer som er blitt benyttet. Estimert levetid vil avhenge av den til enhver tid tilgjengelige datakraft og metodikker som teoretisk kan anvendes for å kompromittere signaturen. Dersom den kryptografiske levetiden for en Elektronisk Signatur anses for å være kortere enn kravet til lagringstid for denne kreves spesielle tiltak for å forlenge signaturens levetid. Et slikt tiltak vil som oftest resultere i at ny Valideringsdata (f.eks. et Tidsstempel) opprettes, informasjon som også vil måtte tas vare på og gjøres tilgjengelig for validering i ettertid.

I en nasjonal kontekst vil Elektroniske Signaturer oppstå gjennom ulike typer av samhandling mellom aktører og med ulike Tillitstjenesteleverandører (eks. tilbydere av valideringstjenester og Sertifikatutstedere) involvert. Signaturen med de nødvendig Valideringsdata skal kunne utveksles og valideres av vilkårlige aktører i ettertid.

Enhetlige løsninger for lagring av elektroniske signert materiale og tilhørende Valideringsdata vil bidra til å forenkle tilgangen til denne informasjonen for aktører som har behov for å foreta signaturvalidering.

Formålet med dette dokumentet er å spesifisere et dataobjekt som skal kunne benyttes for lagring og utveksling av Elektroniske Signaturer, herunder Kvalifiserte Signaturer, som skal kunne valideres over tid. Spesifikasjonen skal danne et best mulig grunnlag for harmonisering av løsninger fra ulike aktører i det norske markedet.

SEID-SDO skal blant annet kunne anvendes ved arkivering av Elektroniske Signaturer i tråd med relevante krav som stilles i §26 i Forskrift om elektronisk kommunikasjon med og i forvaltningen (eForvaltningsforskriften) [20].

[2] Dette dokumentet og løsningen som er anbefalt legger ingen forutsetninger til grunn for hvor lang tid det kan være aktuelt å oppbevare signert materiale.

Målgruppe

Dette dokumentet er primært skrevet for markedsaktører i offentlig og/eller privat sektor som har behov for å utvikle eller anvende løsninger som involverer lagring av Elektroniske Signaturer over tid.

Omfang

Dokumentet spesifiserer et dataobjekt, heretter kalt SEID-SDO (SEID Signert DataObjekt), med definert XML format, informasjonsinnhold og semantikk.

Dataobjektet skal kunne inneholde standardbaserte Elektroniske Signaturer, både CMS [7] formaterte og XMLDSIG [8] formaterte, sammen med nødvendig Valideringsdata som grunnlag for langtidslagring og validering over tid.

SEID-SDO er tilpasset både kortsiktige og mer langsiktige anvendelsesbehov bl.a. gjennom definisjon av ulike SDO profiler og tilhørende minimumskrav til informasjonsinnhold.

I tillegg til relevante internasjonale standarder for signaturformater har et sentralt utgangspunkt for arbeidet vært BankID_SDO [12], som er et tilsvarende dataobjekt utviklet av banknæringen i Norge til bruk innenfor BankID Samarbeidet.

SEID-SDO er et påbygg til ETSI standardene ETSI TS 101 733 [9] og ETSI TS 101 903 [10] og samtidig en videreutvikling av BankID_SDO. SEID-SDO er med andre ord en felles XML struktur som støtter lagring av tre ulike typer signaturobjekter som vist i Figur 3-1.

Avgrensninger

SEID-SDO tilbyr det norske markedet et felles teknisk fundament i form av et XML format for oppbevaring og utveksling av Elektroniske Signaturer med nødvendige Valideringsdata slik at de på enklest mulig måte kan benyttes ved validering over tid. Denne leveransen adresserer derimot ikke generelle utfordringer (eks. av juridisk art) knyttet til langtidslagring av elektronisk informasjon generelt og Elektroniske Signaturer spesielt.

Dokumentet gir ingen føringer når det gjelder hvordan selve signaturgenereringen, som utgangspunkt for opprettelse av et SEID-SDO objekt, skal foregå. Tilsvarende gis det ingen føringer for hvordan signaturvalidering basert på SEID-SDO objektet skal foregå i ettertid.
Konkrete krav til signeringsprosedyrer og til prosedyrer for signaturvalidering reguleres normalt gjennom eksplisitt eller implisitt definerte Signaturpolicyer for den aktuelle anvendelse. Dette dokumentet gir ingen føringer når det gjelder utforming og bruk av Signaturpolicyer.

Når det gjelder Valideringsdata knyttet til en Elektronisk Signatur og som kan legges inn i et SEID-SDO objekt, så er det utenfor leveransens mandat å regulere hvordan Valideringsdata genereres og innhentes.

Konkrete prosedyrer for opprettelse av et SEID-SDO objekt, samt prosesser for oppbevaring/arkivering og vedlikehold av objektet med tanke på langtidslagring ligger også utenfor mandatet.

4.4 Dokumentets struktur

Kapittel 5 beskriver dokumentstatus, overgangsordninger samt prosedyre for dokumentvedlikehold.

Kapittel 6 gir en generell introduksjon til SEID-SDO med blant annet definisjon av en funksjonell referansemodell og en oversikt over objektets interne struktur. Relasjon til eksisterende standarder på området beskrives også.

Kapittel 7 definerer et sett av ulike innholdsprofiler som kan benyttes for et SEID-SDO objekt.

Kapittel 8 definerer et sett av minimumskrav for de profilene som er definert i kapittel 7 og som ikke er dekket i eksisterende standarder.

Bilag A er et normativt bilag og definerer XML skjema for SEID-SDO.

Bilag B er et normativt bilag og beskriver alle elementene som inngår i XML skjemaet i Bilag A.

Bilag C er et informativt bilag som gir en oversikt over standardisering av signaturformater i ETSI.

Bilag D er et informativt bilag og gir en oversikt over noen av de generelle utfordringene som er knyttet til langtidslagring av signaturer og hvordan bruk av et utvidet meldingsformat som SEID-SDO kan bidra til å løse disse.

Bilag E er et informativt bilag som klargjør forholdet mellom SEID-SDO og Kvalifiserte Signaturer.

5 Dokumentets status og forvaltning

5.1 Ansvarlig dokumentforvalter

På oppdrag fra SEID-prosjektet er Post- og teletilsynet utpekt som ansvarlig dokumentforvalter for dette dokumentet. Kontaktpunkt hos Post- og teletilsynet er:

E-post: seid@npt.no

5.2 Status og tilgjengelighet

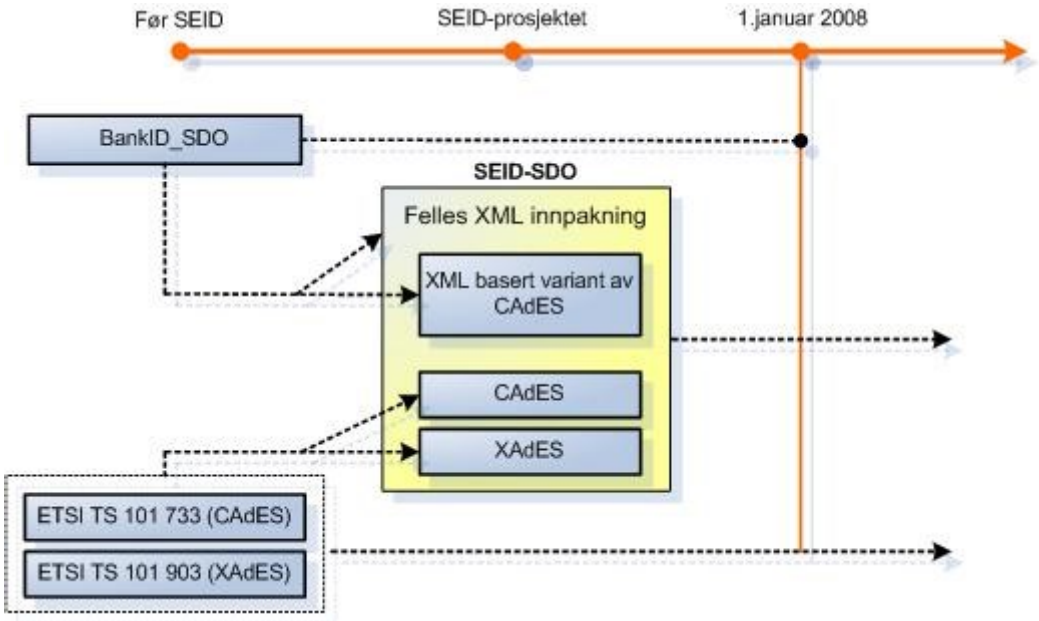
Dette dokumentet definerer et format for lagring av elektroniske signaturer som anbefales benyttet av markedsaktører som har behov for å oppbevare signaturer over tid eller eventuelt utveksle signerte data med andre. Dokumentet er ikke en standard, men en omforent spesifikasjon fra aktørene som har deltatt i SEID-prosjektet.

Dokumentet inneholder kun offentlig informasjon og kan distribueres fritt.

5.3 Overgangsordninger

Enkelte av SEID-prosjektets aktører har allerede operative løsninger i markedet hvor det anvendes dataobjekter for lagring av signaturer som ikke følger SEID-SDO. Dette gjelder spesielt banknæringen i Norge som gjennom BankID Samarbeidet benytter BankID_SDO [12]. For å ivareta dette er det definert en overgangsperiode frem til 31.desember 2007 hvor BankID_SDO likestilles med SEID-SDO når det gjelder lagring og utveksling av CMS signaturer[3]. Fra og med 1.januar 2008 vil BankID_SDO [12] ikke være konform med denne spesifikasjonen fra SEID-prosjektet.

I tillegg til håndteringen av BankID_SDO gir mandatet for denne leveransen i kap. 4.2 en føring som sier at signaturobjekter som følger ETSI TS 101 733 og ETSI TS 101 903 også skal kunne benyttes utenfor det felles rammeverket som SEID-SDO tilbyr. Figur 5-1 viser hvordan denne bruken av disse standardene likestilles med bruk av SEID-SDO, også etter 1. januar 2008.



Figur 5-1 – Overgangsordning for BankID_SDO

Som et påbygg til ETSI standardene tilbyr SEID-SDO en enhetlig XML struktur og således et felles utvekslingsformat som favner både ETSI konforme signaturobjekter samt et signaturobjekt som er en videreutvikling av BankID_SDO. Til bruk for utveksling av signert materiale i Norge anbefales bruk av SEID-SDO fremfor bruk av ETSI standardene alene.

Uansett vil markedet vil måtte forholde seg til at det vil kunne lagres og utveksles ulike typer signaturobjekter, også etter 1. januar 2008. Dette dokumentet regulerer ikke krav til valideringsaktører i forhold til hvilke objekter som skal kunne valideres på et gitt tidspunkt og i gitte kontekster fremover i tid.

[3] Lagring av XMLDSIG formaterte signaturer omfattes ikke av BankID_SDO.

5.4 Dokumentvedlikehold

Aktørene i SEID-prosjektet har i fellesskap besluttet at dette dokumentet skal kunne revideres ved behov og at en behovsvurdering skal gjennomføres årlig.

Ansvarlig dokumentforvalter (se kap. 5.1) er kontaktpunkt for eventuelle spørsmål og konkrete endringsforslag til innholdet i dette dokumentet.

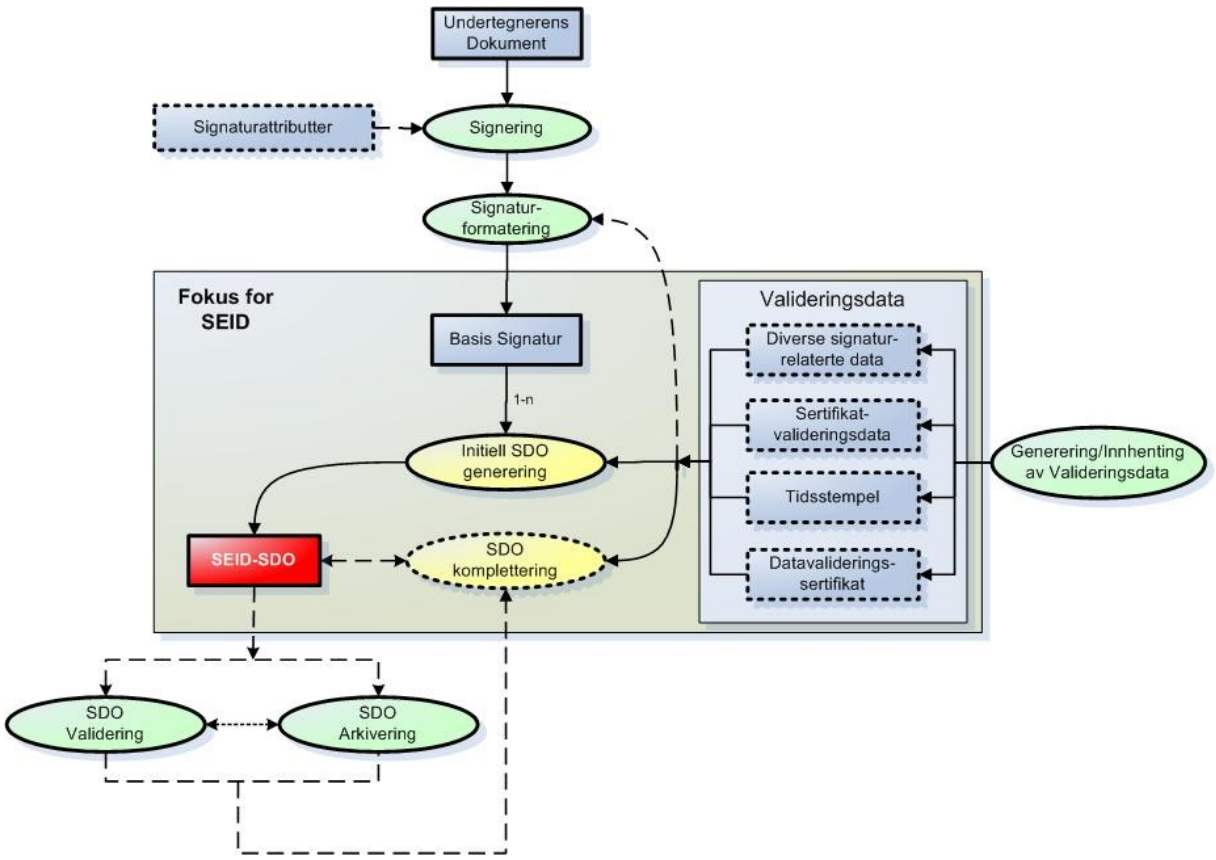
Prosedyrer for revisjon og dokumentvedlikehold er regulert gjennom en egen forvaltningsinstruks utarbeidet av SEID-prosjektet.

6 SEID-SDO

6.1 Hva er SEID-SDO?

SEID-SDO er et dataobjekt definert ved et XML format (se XML skjema i Bilag A) og nødvendig semantikk for at objektet og informasjonsinnholdet skal kunne anvendes og forstås av vilkårlige parter. Dataobjektet skal kunne oppbevare standardbaserte Elektroniske Signaturer og tilhørende Valideringsdata som grunnlag for langtidslagring og validering over tid.

6.2 Referansemodell



Figur 6-1 – Referansemodell

Figur 6-1 viser overordnet hvilke prosesser (ovaler) og informasjonselementer (firkanter) som er relevant ved oppbygging og anvendelse av et SEID-SDO objekt. Informasjonselementene angitt i figuren er nærmere definert i kap.1 mens Tabell 6-1 gir en nærmere beskrivelse av de angitte prosessene.

Hovedfokus i denne leveransen er prosesser og informasjonselementer innenfor det skraverte området i figuren da disse har mer eller mindre direkte relevans for å opprette et SEID-SDO objekt. Føringer knyttet til gjennomføringen av prosessene utenfor det skraverte området anses derimot å ligge utenfor denne leveransens mandat.

Undertegnerens Dokument kan være et hvilket som helst dataobjekt med et definert format, f.eks. et PDF formatert dokument, en ASCII tekststreng, et XML formatert SEID-SDO objekt osv. For å gi signaturen økt sikkerhetsverdi med tanke på signaturvalidering i ettertid er det vanlig at signeringen ikke foretas over Undertegnerens Dokument alene, men at enkelte andre Signaturattributter inkluderes. Vi snakker i dette tilfelle om en signatur med multiple signerte attributter. Eksempler på slike Signaturattributter er referanse til Undertegnerens Signatursertifikat[4], angivelse av hvilken Signaturpolicy signeringen er foretatt under, tidsangivelse for signering, et Tidsstempel[5] over Undertegnerens dokument osv.

Prosess	Prosessbeskrivelse
Signering	Prosess hvor en Undertegner foretar en kryptografisk signeringsoperasjon ihht. aktuell Signaturpolicy. Signeringen skjer over Undertegnerens Dokument og eventuelt nærmere angitte Signaturattributter. Nødvendig sammenstilling og formatering av disse dataene i forkant av selve signeringen er en del av denne prosessen. Resultat fra signeringsprosessen vil være en dataverdi som representerer signaturen (signaturverdi).
Signaturformatering	Prosess for å pakke signaturverdien fra foregående prosess inn i en Basis Signatur som vil ha en standardisert intern formatstruktur i henhold til CMS [7] eller XMLDSIG [8]. I tillegg vil Valideringsdata[6] kunne legges inn som usignerte attributter i den samme strukturen (angitt med stiplet pil i Figur 6-1).
Initiell SDO generering	Prosess for å opprette et nytt SEID-SDO objekt som inneholder én eller flere Basis Signaturer og tilhørende Valideringsdata knyttet til hver disse. Ethvert SEID-SDO objekt må innholdsmessig tilfredsstille et sett av minimumskrav i forhold til en nærmere valgt profil (se kap.7).
SDO komplettering	Prosess for å komplettere et eksisterende SEID-SDO objekt ved å legge inn ytterligere Valideringsdata. Typisk vil en slik prosess benyttes for å forlenge levetiden på de enkelte Basis Signaturene, f.eks. ifm. langtidslagring. Det kompletterte objektet må innholdsmessig tilfredsstille et sett av minimumskrav i forhold til en nærmere valgt profil (se kap.7).

[4] Er sentralt i forhold Lov om elektronisk signatur [16] som krever at en Avansert Elektronisk Signatur skal være entydig knyttet til Undertegneren.
[5] Benyttes som bevis for at man var i besittelse av et dokument på et gitt tidspunkt.
[6] Hvilke Valideringsdata som evt. inkluderes kan variere fra løsning til løsning.

Prosess	Prosessbeskrivelse
Generering/Innhenting av Valideringsdata	Prosess for å skaffe til veie, evt. å få generert, nødvendige Valideringsdata for innlegging i et SEID-SDO objekt. Hvilke dataelementer det er behov for er avhengig av ønsket profil og de policymessige føringer som er gitt av den aktuelle anvendelse.
SDO Validering	Prosess for å validere et eksisterende SEID-SDO objekt. I enkelte tilfeller kan innhenting av eksterne Valideringsdata være nødvendig for å komplettere de Valideringsdata som måtte befinne seg i selve SEID-SDO objektet.
SDO Arkivering	Prosess for å lagre og oppbevare eksisterende SEID-SDO objekter over kortere eller lengre tid.

Tabell 6-1 – Prosesser i referansemodellen

6.3 Overordnede krav til SEID-SDO

Punktlisten nedenfor angir noen av de sentrale krav som har vært førende for spesifikasjonen av SEID-SDO:

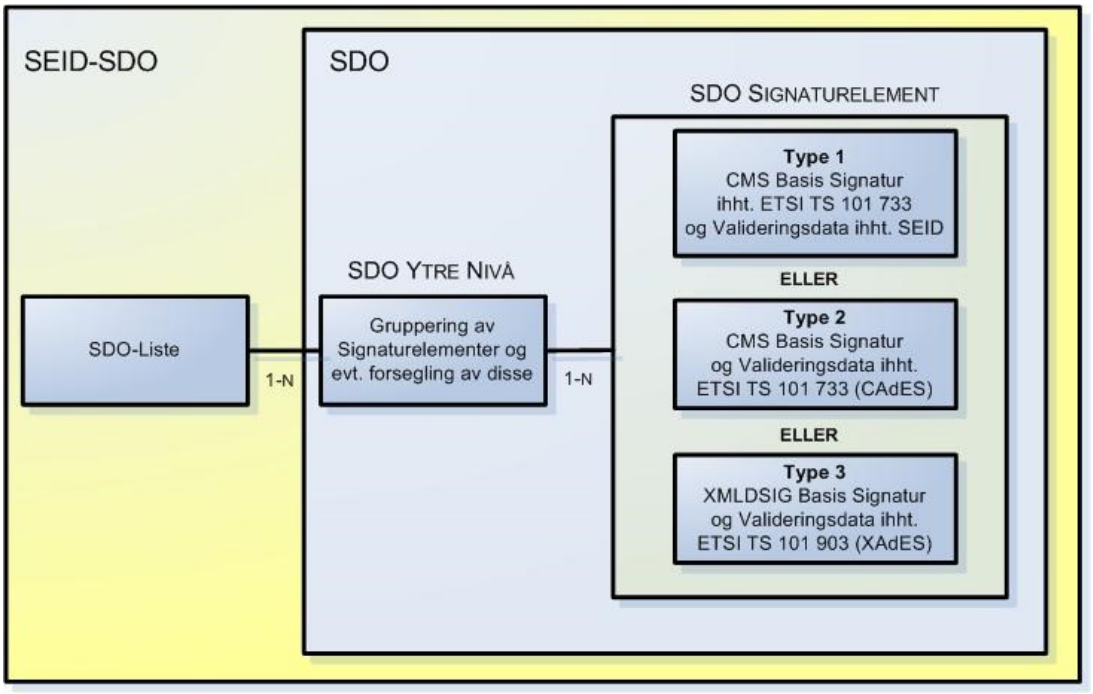
- SEID-SDO skal ikke være til hinder for at et SEID-SDO opprettet gjennom én samhandling senere skal kunne anvendes i andre samhandlinger med andre utøvere og deres Tillitstjenesteleverandører.
- SEID-SDO skal ikke være til hinder for at utøvende parter i en samhandling fritt skal kunne velge ønskede Tillitstjenesteleverandører.
- SEID-SDO skal ha plassholdere for de data som er relevant for signaturvalidering og for integritetsbeskyttelse av disse.
- SEID-SDO skal være fremtidsrettet og fleksibelt i forhold til hvilken informasjon som kan legges inn, og hvor og av hvem objektene kan lages.
- SEID-SDO skal ta hensyn til anerkjente internasjonale standarder.
- SEID-SDO skal ta hensyn til BankID_SDO [12] og skal sikre innholdsmessig kompatibilitet.
- SEID-SDO skal gi involverte parter størst mulig grad av bevisverdi.
- SEID-SDO skal være praktisk å lagre, utveksle og validere.
- SEID-SDO skal være praktisk i bruk, både for eksisterende behov, og for senere behov.
- SEID-SDO skal støtte lagring av både CMS [7] formaterte signaturer og XMLDSIG [8] formaterte signaturer.
- SEID-SDO skal ikke være begrensende med tanke på hvilket format Undertegnerens Dokument skal kunne ha.
- SEID-SDO skal kunne fungere både som et lagringsformat og som et utvekslingsformat for Elektroniske Signaturer som skal oppbevares over tid.

6.4 SEID-SDO og formatstandardisering

ETSI har utarbeidet to ulike formatstandarder for Elektroniske Signaturer som skal ha gyldighet over tid (se Bilag C for detaljer);

- ETSI TS 101 733 (CAdES) [9] som formatmessig bygger på CMS [7].
- ETSI TS 101 903 (XAdES) [10] som formatmessig bygger på XMLDSIG [8].

Det finnes også en standard [12] for banknæringen i Norge som beskriver et dataobjekt (BankID_SDO) for lagring av signerte data til bruk innenfor BankID Samarbeidet. Dette formatet er XML basert og støtter lagring av CMS [7] formaterte signaturer.



Figur 6-2 – SEID-SDO struktur

Figur 6-2 viser hvordan et SEID-SDO objekt er bygget opp. På innerste nivå i SEID-SDO finner vi SDO Signaturelementer. Tre ulike typer Signaturelementer er definert:

- **Type 1:** En XML struktur definert av SEID-prosjektet for lagring av en CMS formatert Basis Signatur og tilhørende Valideringsdata. Utgangspunktet for denne typen Signaturelement er ETSI TS 101 733 og BankID_SDO [12].
- **Type 2:** Ett XML element for lagring av et CMS objekt i henhold til ETSI TS 101 733 [9], dvs. at objektet både oppfyller standardens krav til en Basis Signatur og til eventuell lagring av tilhørende Valideringsdata.
- **Type 3:** En XML struktur i henhold til XAdES [10] for lagring av en XMLDSIG [8] formatert Basis Signatur og tilhørende Valideringsdata.

Hovedforskjellen mellom Signaturelementer av Type 1 og 2 over er at Type 1 åpner for at Valideringsdata kan plasseres i egne XML elementer utenfor selve CMS objektet mens ETSI TS 101 733 og Type 2 krever at Valideringsdataene plasseres i ASN.1 attributter som en del av selve CMS objektet.

Et SDO Signaturelement pakkes inn i en ytre XML struktur hvorav denne helheten utgjør et SDO objekt. Det er mulig å samle flere SDO Signaturelementer i samme SDO objekt. Forutsetningen er at alle Signaturelementene i SDO objektet inneholder en Elektronisk Signatur som er generert over det samme dataobjektet, dvs. ”Undertegnerens Dokument” i Figur 6-1. Den ytre XML strukturen gir videre mulighet til å forsegle og dermed integritetsbeskytte alle SDO Signaturelementene i objektet som én enhet.

SEID-SDO gir også mulighet til å gruppere ett eller flere SDO objekter i én SDO-liste. Dersom en samhandling mellom to parter resulterer i flere signaturer over ulike dataobjekter

(Undertegnerens Dokument) vil disse måtte lagres i avhengige SDO objekter. En SDO liste kan således brukes til å samle ulike SDO objekter som naturlig hører sammen.

7 SEID-SDO profilering

Den profilering av signaturobjekter som er foretatt av ETSI (se Bilag C) er analogt med en profilering av det enkelte SDO Signaturelement i SEID-SDO Figur 6-2. Den ytre XML strukturen i SEID-SDO er et ekstra påbygg basert på BankID_SDO og omfattes ikke av ETSI spesifikasjonene.

For å ivareta en enklest mulig kobling mellom SEID-SDO og ETSI arbeidet er det derfor valgt å håndtere profilering av SDO Signaturelementer og den ytre XML strukturen (SDO ytre nivå) hver for seg.

7.1 SDO Signaturelementer

7.1.1 SDO Signaturelementer – Type 1

I arbeidet med å definere relevante profiler for SDO Signaturelementer av Type 1 (se Figur 6-2 i kap. 6.4) har det vært viktig med størst mulig grad av harmonisering med de profilene som ETSI har definert. Samtidig har det vært et mål å foreta forenklinger der det har vært mulig i lys av at ETSI standardene er forholdsvis generelle, se kapittel 7.1.1.1.

For SDO Signaturelementer Type 1 er det definert seks forskjellige profiler som vist i Tabell 7-1. Tabellen viser også hvordan de ulike profilene formålsmessig relaterer seg til tilsvarende profiler definert av ETSI. Den enkelte profil er forbundet med et sett av minimumskrav til informasjonsinnhold i Signaturelementet, se kap.8 for detaljer. Ett enkelt SDO objekt kan inneholde flere SDO Signaturelementer og hver av disse kan profileres uavhengig av hverandre.

Profilnavn	Profilbeskrivelse	Analogi til ETSI profil (se Bilag C)	Formål med profilen
SEID-SDO-Basic	Minimumsprofil med Basis Signatur og evt. et minimums sett av obligatoriske Valideringsdata.	ES (BES/EPES)	Lagring av Basis Signatur uten Valideringsdata.
SEID-SDO-Basic-V	SEID-SDO-Basic + ytterligere krav til Sertifikatvalideringsdata.	Omfattes ikke av ETSI	Lagring av Basis Signatur med basis Valideringsdata. Er innført for å dekke behov innenfor BankID Samarbeidet.
SEID-SDO-Basic-T	SEID-SDO-Basic + krav om tidsstemplet signaturverdi.	ES-T	Se Bilag D.
SEID-SDO-TSP	SEID-SDO-Basic + krav om Datavalideringssertifikat utstedt av en Tillitstjenesteleverandør.	Omfattes ikke av ETSI	Alternativ til bruk av SEID-SDO-Extended dersom man benytter signaturvalidering som en 3.parts tillitstjeneste.

Profilnavn	Profilbeskrivelse	Analogi til ETSI profil (se Bilag C)	Formål med profilen
SEID-SDO-Extended	SEID-SDO-Basic-T + krav om komplette Valideringsdata knyttet til den aktuelle Basis Signatur.	ES-X-Long	Se Bilag D.
SEID-SDO-Archive	SEID-SDO-Basic + krav om Tidsstempel over både signaturverdi og komplette Valideringsdata.	ES-A	Se Bilag D.

Tabell 7-1 – Profilering av SDO Signaturelementer

Profilene i Tabell 7-1 er definert ut ifra tanken om at det for en gitt Basis Signatur skal være mulig å starte med et SDO Signaturelement i henhold til en av minimumsprofilene (SDO-Basic, SDO-Basic-V eller SDO-Basic-T) for deretter å utvide denne til en av de mer omfattende profilene over tid. Kapittel 8 definerer et sett av minimumskrav til informasjonsinnhold for den enkelte profil i tabellen.

7.1.1.1 Forenklinger i forhold til definerte ETSI profiler

ETSI standardene (se Bilag C) anses for å være veldig generelle. Ved definisjon av profilene i Tabell 7-1 er derfor følgende forenklinger foretatt i forhold til ETSI TS 101 733:

- Profilen ETSI ES-C hvor Valideringsdata inkluderes gjennom utelukkende bruk av referanser anses å ha liten praktisk verdi og er derfor ikke videreført som egen profil. Dersom Valideringsdata ønskes inkludert i SDO Signaturelementet, anses det formålstjenlig å kreve innlegging av de faktiske Valideringsdata, bl.a. for å forenkle tilgangen til disse ved signaturvalidering i ettertid. Bruk av referanser til Valideringsdata er kun videreført som en opsjon for CRLer. Dette for å kunne spare lagringsplass i og med at enkelte CRLer over tid kan bli forholdsvis omfattende i størrelse.
- Ved at SDO Signaturelement Type 1 ikke definerer en egen ES-C profilvariant gir det tilsvarende lite mening å videreføre ES-X-Type-1, dvs. Tidsstempel over ES-C data, og ES-X-Type-2, dvs. Tidsstempel over referanser til Valideringsdata. Av den grunn er ETSI profilene ES-X-Type-1, ES-X-Type-2 samt ES-X-Long-Type1 og ETSI ES-X-Long-Type2 ikke videreført som egne profiler.
- ETSI ES-A profilen krever at en tidsstemplet signaturverdi (ES-T) må foreligge før ES-A påføres. For SEID-SDO-Archive er dette kravet ikke videreført. Tidsstempelet knyttet til SEID-SDO-Archive kan påføres uten at SEID-SDO-Basic-T profilen ligger til grunn.

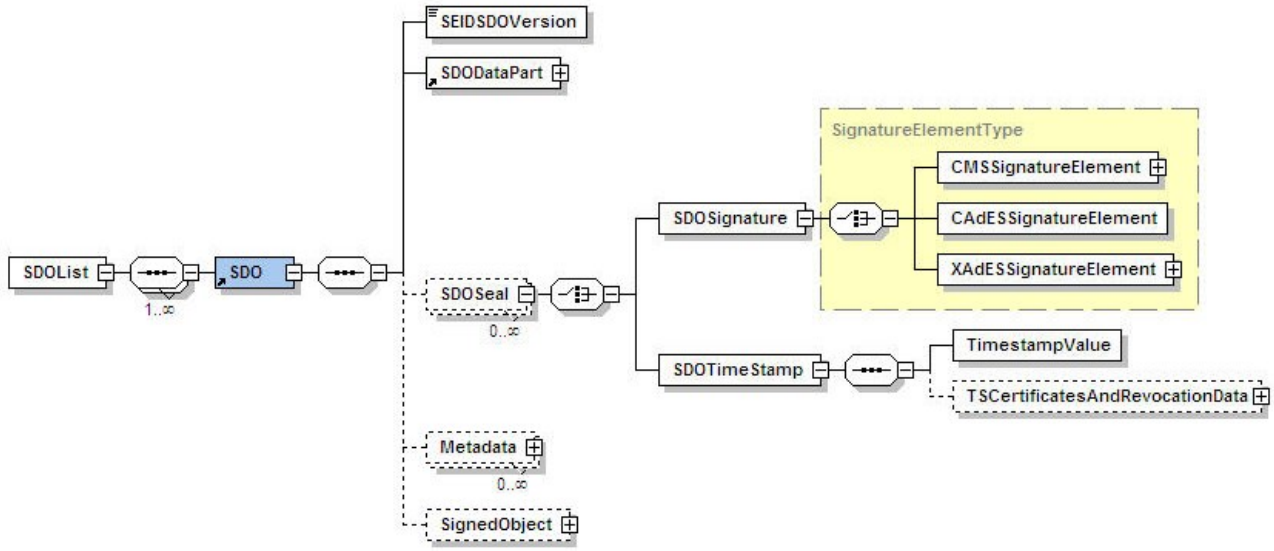
7.1.2 SDO Signaturelementer – Type 2 og 3

For SDO Signaturelementer av Type 2 og 3 (se kap. 6.4) er det lagt til grunn at disse skal følge ETSI standardene TS 101 733 og TS 101 903 og de basisprofilene som disse definerer (se Bilag C) fullt ut.

En potensiell utfordring med ETSI profilene når det gjelder praktisk anvendelse er at de er veldig generelle. En eventuell subprofilering av ETSI standardene vil kunne bidra til å snevre inn mulighetsrommet noe. Det ligger derimot utenfor SEID-prosjektets mandat å foreta en slik subprofilering.

På generelt grunnlag anses profiltankegangen for SDO Signaturelementer av Type 1 i kap. 7.1 og de forenklinger i forhold til ETSI som er beskrevet i kap. 7.1.1.1 å være et godt utgangspunkt for å tenke en eventuell subprofilering SDO Signaturelementer av Type 2 og 3. På den annen side virker ETSI standardene og de basisprofilene disse definerer forholdsvis rigide slik at de samme forenklingene ikke uten videre kan videreføres dersom man skal holde seg innenfor rammene av det ETSI standardene tillater. For eksempel virker det som om ETSI krever at profilene ES-X-Long og ES-A må være basert på profilen ES-C, en profil som man for Type 1 SDO Signaturelementer har valgt å se bort fra.

7.2 SDO ytre nivå



Figur 7-1 – SDO ytre nivå

Figur 7-1 viser XML strukturen som er valgt for SDO ytre nivå. Det er videre definert fire ulike forseglingsprofiler for SDO ytre nivå som vist i Tabell 7-2. Disse profilene angir om, og i tilfelle hvordan, SDO Signaturelementene som ligger under elementet SDODatapart som helhet er forseglet.

Profilnavn	Profilbeskrivelse
SDO-Unsealed	SDO uten signatur eller Tidsstempel påført på ytre nivå.
SDO-CMSSigned	CMS signatur [7] generert over elementet SDODataPart i SDO objektet (se Figur 7-1). I dette tilfellet benyttes elementet SDOSignature med ett av underelementene CMSSignatureElement eller CADESSignatureelement til å lagre forseglingssignaturen og eventuelt tilhørende Valideringsdata.
SDO-XMLSigned	Detached type XMLDSIG signatur generert over elementet SDODataPart i SDO objektet (se Figur 7-1). I dette tilfellet benyttes elementet SDOSignature med underelement XAdESSignatureElement til å lagre forseglingssignaturen og eventuelt tilhørende Valideringsdata.
SDO-Timestamped	Tidsstempel generert over elementet SDODataPart i SDO objektet (se Figur 7-1). I dette tilfellet benyttes elementet SDOTimestamp til å lagre Tidsstempelet og eventuelt tilhørende Valideringsdata.

Tabell 7-2 – Profilering av SDO ytre nivå

Det eneste som skiller de fire forseglingsprofilene i Tabell 7-2 er bruken av XML elementet SDOSeal. Bruken av øvrige XML elementer på SDO ytre nivå er ikke avhengig av hvilket av forseglingsalternativene som anvendes.

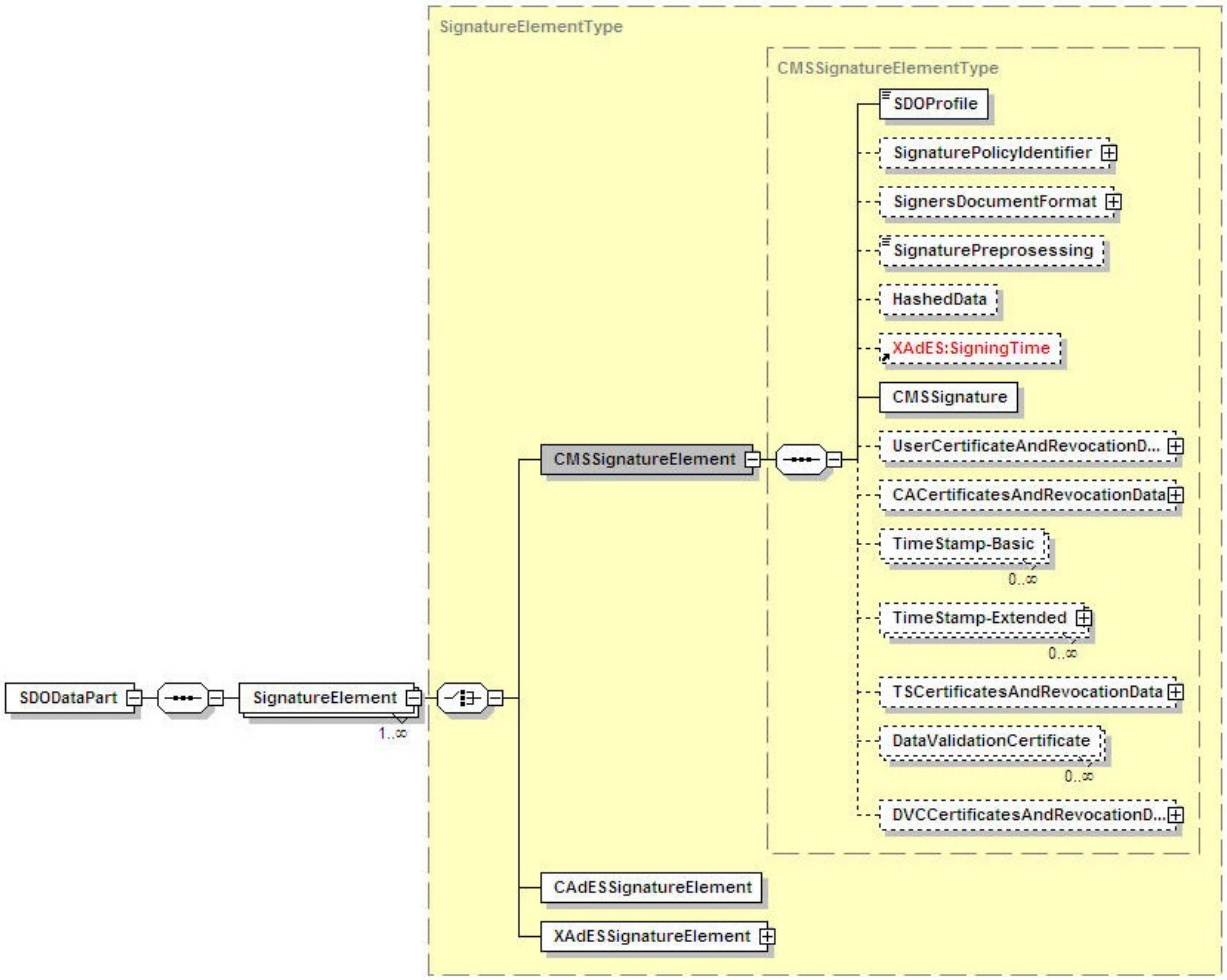
Kombinasjoner av profiler på SDO Signaturelement nivå og på SDO ytre nivå kan i utgangspunktet velges fritt og tilpasses den enkelte anvendelse.

7.2.1 Produksjon av tidsstempel for SDO-Timestamped

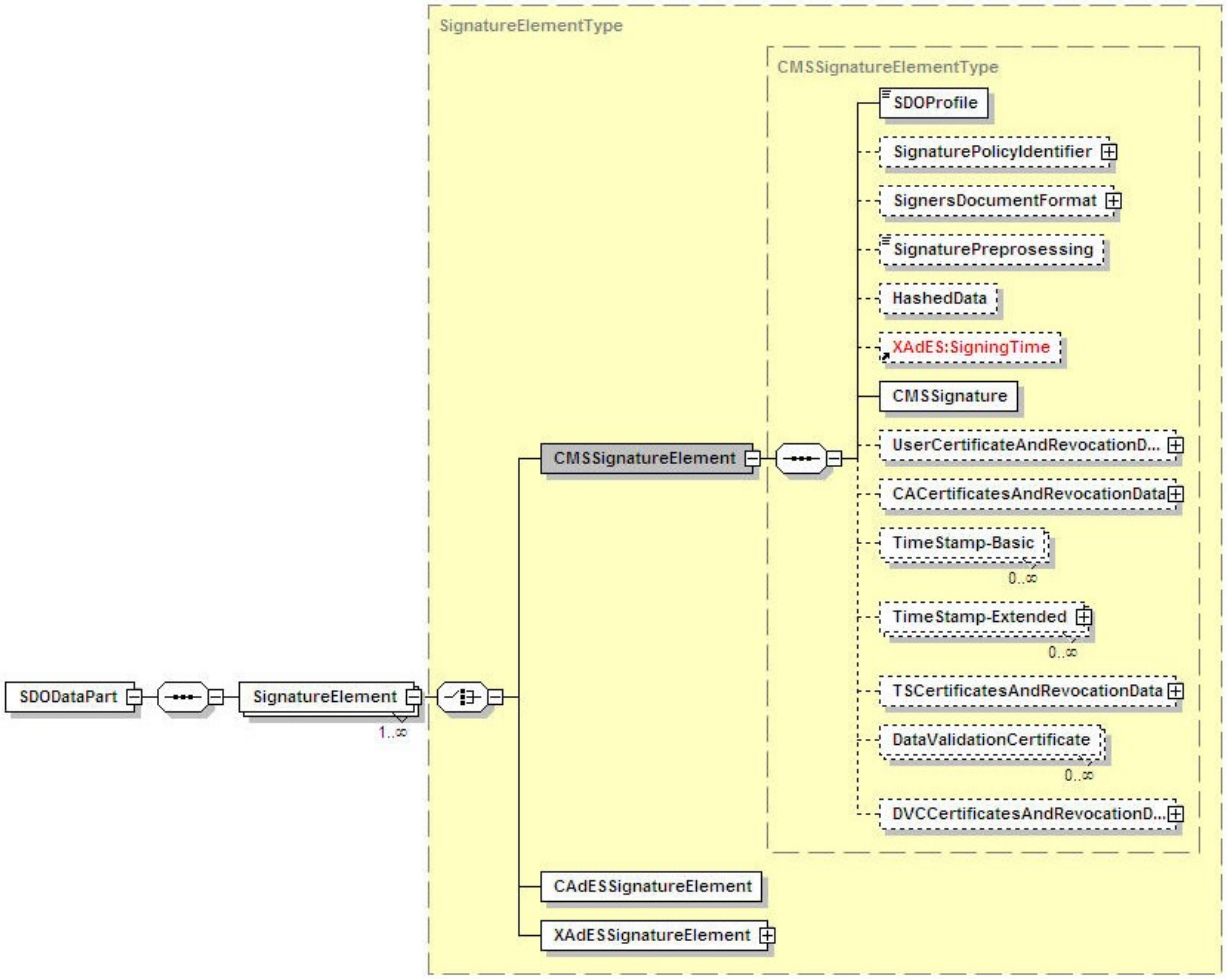
Se kap. 8.2.3 for en beskrivelse av den preprosessering som er nødvendig for å produsere det dataelementet som skal benyttes som input til tidsstempling. Når det gjelder bruk av Include elementer for SDO-Timestamped (ref. kap. 8.2.3) er det nok å benytte ett slikt element hvor URI attributtet refererer elementet SDODataPart.

Det ligger utenfor mandatet til denne leveransen å sette krav til internt format for Tidsstempel. Generelt anses formatet definert i RFC 3161 [14] og evt. ETSI TS 101 861 [18] som naturlig å benytte.

8 Profilkrav for SDO Signaturelementer av Type 1



Figur 8-1 – SDO Signaturelement Type 1



Figur 8-1 viser XML strukturen CMSSignatureElementType som gjelder for SDO Signaturelementer Type 1. For eventuelle XML understrukturer under de elementene som er vist i figuren henvises det til Bilag A og Bilag B. Tabell 8-1 viser hvilke minimumskrav til informasjonsinnhold som er definert for den enkelte profil definert i kap. 7.1.1. I tabellens andre kolonne er følgende forkortelser benyttet for å angi type krav:

- ”M” – Må krav
- ”B” – Betinget krav
- ”O” – Opsjon

Profil	Type krav (M/B/O)	Krav til hvilke XML elementer som skal benyttes	
SEID-SDO-Basic	**M**	• SDOProfile • CMSSignature • SignersDocumentFormat Det stilles også egne krav knyttet til Basis Signaturen som lagres i elementet CMSSignature. Se kap.8.1 for detaljer.	
	B	• HashedData	M-krav dersom Undertegnerens Dokument er hashet én gang før CMS signaturen påføres, se kap. 8.1.1. Elementet skal ikke benyttes ellers.
	B	• SigningTime	M-krav dersom CMS Signaturen er påført over både Undertegnerens Dokument og angitt tid, se kap. 8.1.1. I øvrige tilfeller er bruk av elementet opsjonelt.
	B	• SignaturePreprosessing	Dersom preprosessering av Undertegnerens Dokument er foretatt i forkant av CMS signering skal elementet SignaturePreprosessing benyttes for å angi dette. Se kap. 8.1.1.
	O	• UserCertificateAndRevocationData • CACertificatesAndRevocationData • SignaturePolicyIdentifier	
SEID-SDO-Basic-V	**M**	• Alle M-krav som for SEID-SDO-Basic • UserCertificateAndRevocationData hvor OCSP respons og/eller CRL knyttet til Undertegnerens signeringssertifikat er lagt inn	
	B	• Alle B-krav som for SEID-SDO-Basic	
	O	• Alle O-krav som for SEID-SDO-Basic	
SEID-SDO-Basic-T	**M**	• Alle M-krav som for SEID-SDO-Basic • TimeStamp-Basic (se kap. 8.2.1**Feil! Fant ikke referansekilden.**)	
	B	• Alle B-krav som for SEID-SDO-Basic	
	O	• Alle O-krav som for SEID-SDO-Basic • TSCertificatesAndRevocationData med sertifikater og revokeringsinformasjon knyttet til TimeStamp-Basic	
SEID-SDO-TSP	**M**	• Alle M-krav som for SEID-SDO-Basic • DataValidationCertificate med et Datavalideringssertifikat knyttet til CMS signaturen i CMSSignature.	
	B	• Alle B-krav som for SEID-SDO-Basic	
	O	• Alle O-krav som for SEID-SDO-Basic • TimeStamp-Basic • TSCertificatesAndRevocationData med sertifikater og revokeringsinformasjon knyttet til TimeStamp-Basic • DVCCertificatesAndRevocationData som inneholder: o Alle CA sertifikater i tillitskjede for Datavalideringssertifikatet. o Revokeringsdata (typisk OCSP responser og/eller CRLer) knyttet til ovennevnte CA sertifikater.	
SEID-SDO-Extended	**M**	• Alle M-krav som for SEID-SDO-Basic-T • UserCertificateAndRevocationData som inneholder både o Undertegnerens Signatursertifikat. o Revokeringsdata (typisk OCSP respons og/eller CRL) knyttet til dette sertifikatet. • CACertificatesAndRevocationData som inneholder: o Alle CA sertifikater i tillitskjede for Undertegnerens Signatursertifikat. o Alle CA sertifikater i tillitskjede for eventuell OCSP responder. o Revokeringsdata (typisk OCSP responser og/eller CRLer) knyttet til ovennevnte CA sertifikater.	
	B	• Alle B-krav som for SEID-SDO-Basic	
	O	• SignaturePolicyIdentifier	

Profil	Type krav (M/B/O)	Krav til hvilke XML elementer som skal benyttes
SEID-SDO-Archive	M	• Alle M-krav som for SEID-SDO-Basic • UserCertificateAndRevocationData og CACertificatesAndRevocationData ihht. SEID-SDO-Extended eller DataValidationCertificate ihht. SEID-SDO-TSP • Timestamp–Extended (se kap. 8.2.2)
	B	• Alle B-krav som for SEID-SDO-Basic
	O	• TimeStamp-Basic • TSCertificatesAndRevocationData med sertifikater og revokeringsinformasjon knyttet til TimeStamp-Extended og/ellerTimeStamp-Basic • DVCCertificatesAndRevocationData med sertifikater og revokeringsinformasjon knyttet til DataValidationCertificate dersom dette elementet benyttes. • SignaturePolicyIdentifier

Tabell 8-1 – Profilkrav for SDO Signaturelementer av Type 1

8.1 Krav til Basis Signatur

Basis Signaturen skal være en CMS (PKCS#7[7]) signatur ihht. [7]. I ETSI TS 101 733 defineres to alternative profiler for en slik Basis Signatur, BES (*Basic Electronic Signature*) og EPES (*Explicit Policy-based Electronic Signature*) med tilhørende minimum konformitetskrav.

Kort oppsummert innebærer konformitetskravene for BES at signaturfremstillingssystemet må kunne generere en Basis Signatur som gjør bruk av multiple Signaturattributter i henhold til CMS [7] standarden og hvor en eksplisitt sertifikatreferanse skal være en av disse. De samme kravene gjelder for profilen EPES men i dette tilfelle må også Signaturpolicy inngå som ett av Signaturattributtene.

Etter det SEID-prosjektet kjenner til så vil mobile signaturløsninger[8] som i dag anvendes i det norske markedet ikke uten videre kunne oppfylle de konformitetskravene som ETSI har definert. Resultatet fra signeringsprosessen i en mobilløsning vil typisk enten være en enkel PKCS#1 signatur [17] uten ASN.1 formatering av input data eller en signatur formatert i henhold til WAP SignedContent[9] [15]. Førstnevnte løsning vil ha problemer med å støtte multiple Signaturattributter ihht. CMS standarden [7] i det store og hele mens sistnevnte løsning vil ha problemer med å støtte signaturreferanse og evt. Signaturpolicy som Signaturattributter spesielt. Disse begrensningene er hensyntatt når kravene knyttet til Basis Signaturen er definert i Tabell 8-2.

#	Krav	Kommentar
1	Basis Signaturen skal være CMS formatert ihht.[7]	
2	De CMS relaterte konformitetskrav som ETSI TS 101 733 har definert for BES, evt. EPES, skal følges dersom den aktuelle signaturløsning teknisk kan støtte disse.	Dersom signaturløsningen ikke støtter bruk av signert sertifikatreferanse (BES/EPES krav) i tråd med CMS standarden anbefales det å finne andre metoder for å skape en entydig og uforfalskelig knytning mellom Basis Signaturen og Undertegnerens sertifikat.

[7] CMS formatet er basert på PKCS#7 versjon 1.5 [21] og er bakover kompatibel med denne.
[8] SIM Toolkit baserte signaturløsninger.
[9] Merk at bruk av WAP SignedContent ikke er bundet til bruk av WAP som kommunikasjonsprotokoll.

#	Krav	Kommentar
3	Basis Signaturen (CMS objektet) skal kun inneholde én enkelt signatur fra én enkelt Undertegner.	CMS standarden åpner for lagring av multiple sidestilte signaturer og/eller kontrasignaturer i samme CMS objekt. For SEID-SDO skal evt. multiple signaturer håndteres som flere CMS objekter og lagres i ulike SDO Signaturelementer.
4	Valideringsdata som både finnes som signerte og/eller usignerte attributter i CMS objektet samt som egne XML informasjonselementer i SEID-SDO objektet skal være i samsvar.	SEID-SDO objektet er ugyldig dersom det ikke er samsvar.

Tabell 8-2 – Krav til Basis Signatur for SDO Signaturelementer av Type 1

8.1.1 Preprosessering av inputdata til signeringsprosessen

For at en aktør skal kunne validere knytningen mellom Undertegnerens Dokument og en tilhørende Basis Signatur er det helt nødvendig at aktøren ut ifra informasjon i SDO objektet vet hvordan Undertegnerens Dokument og eventuelle ekstra Signaturattributter (Figur 6-1) er preprosessert i forkant av signeringen. To metoder er aktuelle i denne kontekst:

- **Uten preprosessering:** Undertegnerens Dokument og eventuelle øvrige Signaturattributter håndteres i henhold til CMS standarden uten ytterligere preprossesering i forkant. I dette tilfellet vil Undertegnerens Dokument og de enkelte Signaturattributtene havne som separate attributter i CMS objektet og CMS objektet alene vil inneholde nok informasjon til at mottaker vet hvordan dette skal tolkes.
- **Med preprosessering:** Undertegnerens Dokument og eventuelle Signaturattributter preprosesseres og settes sammen til ett nytt dataobjekt i forkant av CMS signering. Dataobjektet vil havne som ett enkelt attributt i CMS objektet og CMS objektet i seg selv vil ikke inneholde informasjon om preprosesseringen. Denne informasjonen må i så fall antas implisitt kjent for valideringsaktører i forhold den aktuelle anvendelse, evt. angis eksplisitt gjennom egne XML elementer i SDO objektet.

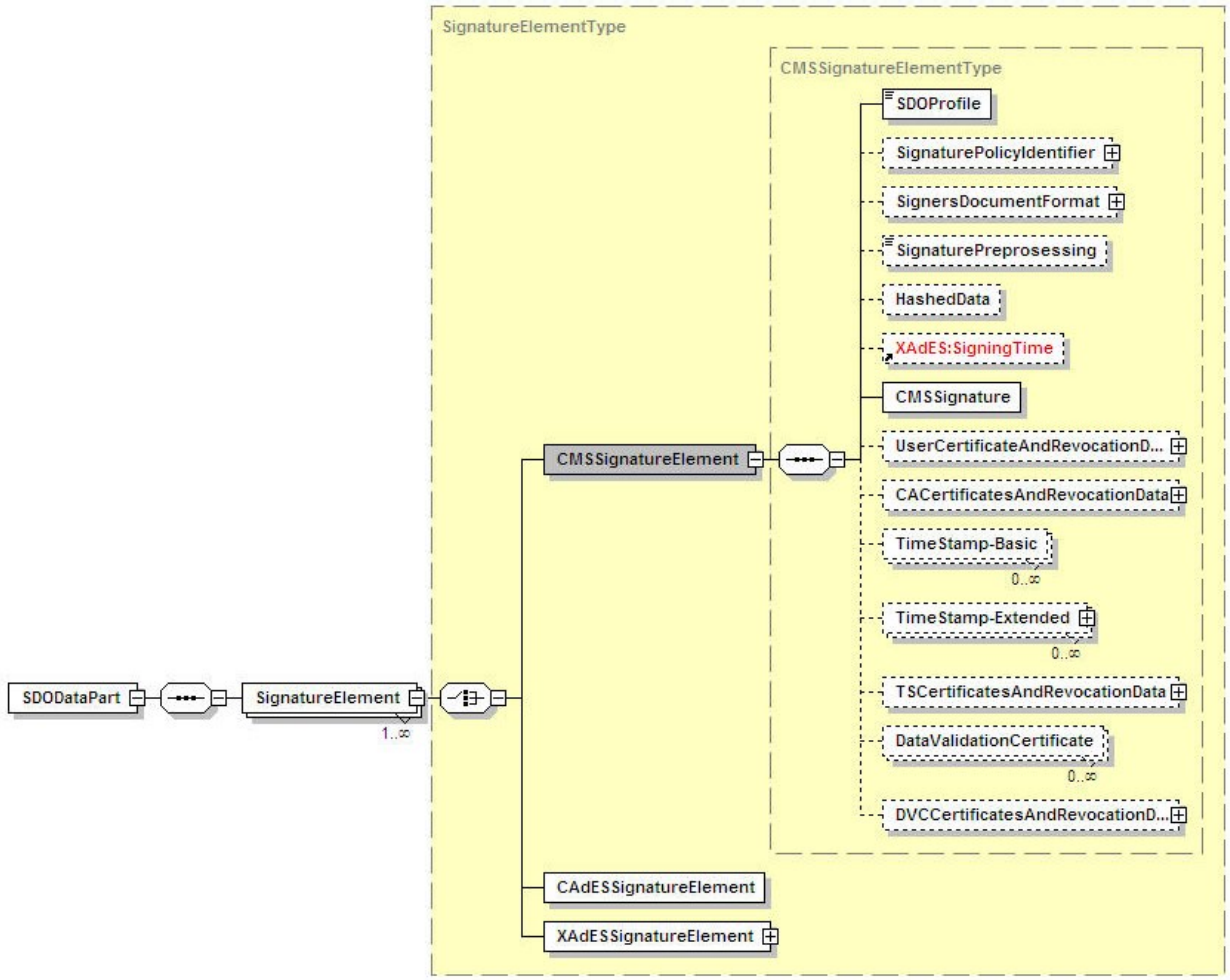
I SEID-SDO er XML elementet SignaturePreprosessing definert for eksplisitt å kunne angi eventuell preprosessering som skjer i forkant av CMS signering. Følgende to alternative preprosesseringsoperasjoner er definert:

1. Inputdata til signeringsprosessen er en hash av Undertegnerens Dokument. I dette tilfellet skal elementet SignaturePreprosessing gis verdien ”Hashed”.
2. Input til signeringsprosessen er en hash av Undertegnerens Dokument konkatenert med dato/tidsangivelse ihht. ISO 8601 [19]. I dette tilfellet skal elementet SignaturePreprosessing gis verdien ”HashedAndTimeAppended”.

8.2 Krav til Tidsstempel

Dette kapittelet beskriver de krav som gjelder for tidstempler generert i tilknytning til profilene SEID-SDO-Basic-T og SEID-SDO-Archive, dvs. Tidsstempel lagret i elementene

TimeStamp-Basic og TimeStamp-Extended i



Figur 8-1.

Kapittelet beskriver kun krav knyttet til hvilke informasjonselementer som skal inngå som input til tidsstemplingen og hvordan disse formateres.

Det ligger utenfor mandatet for denne leveransen å sette krav til internt format på Tidsstempel som genereres. Generelt anses formatet definert i RFC 3161 [14] og evt. ETSI TS 101 861[10] [18] som naturlig å benytte da kommersielle implementasjoner antas å følge disse.

8.2.1 TimeStamp-Basic

Tidsstempelet knyttet til SDO elementet TimeStamp-Basic er et tidstempel over Basis Signaturen som helhet (hele CMS objektet), dvs. et Tidsstempel over dataverdien i SDO elementet CMSSignature. Dette Tidsstempelet vil gi tilsvarende bevisverdi som det Tidsstempelet ETSI definerer gjennom sin ES-T profil selv om det i ES-T tilfellet kun er signaturverdien (dvs. verdien i ett enkelt attributt internt i CMS objektet) og ikke CMS objektet som helhet som tidsstemples.

Se kap. 8.2.3 for en beskrivelse av den preprosessering som er nødvendig for å produsere dataelementet som skal benyttes som input til tidsstempling. Når det gjelder bruk av Include

[10] ETSI TS 101 861 er en videre profilering av RFC 3161.

elementer (ref. kap. 8.2.3) er det nok å benytte ett slikt element hvor URI attributtet refererer elementet CMSSignature.

SDO objektet støtter muligheten for å legge inn flere uavhengige og parallelle Tidsstempel over samme Basis Signatur, dvs, at SDO objektet støtter flere TimeStamp-Basic elementer. Nøstede[11] Tidsstempel på dette nivået støttes ikke. Dette støttes heller ikke av ETSI TS 101 733.

Ved signaturvalidering må Tidsstempelet også kunne valideres. Innlegging av Valideringsdata knyttet til Tidssstempelet er en opsjon. Det er videre valgfritt om eventuelle Valideringsdata legges som usignerte attributter som en del av Tidsstempelets interne datastruktur eller om de legges i elementet TimeStampCertificatesAndRevocationData.

8.2.2 TimeStamp-Extended

Tidsstempelet knyttet til SDO elementet TimeStamp-Extended er et tidstempel påført over både Basis Signaturen samt alle Valideringsdata for det aktuelle SDO Signaturelement. Et TimeStamp-Extended Tidsstempel gir tilsvarende bevisverdi og sikkerhet som det Tidsstempelet som ETSI definerer gjennom sin ES-A profil.

SEID-SDO støtter innlegging av flere Tidsstempel av typen TimeStamp-Extended. I dette tilfellet snakker vi om nøstede tiddstempler, dvs. at eksisterende Timesstamp-Extended type Tidsstempel som allerede befinner seg i SDO Signaturelementet vil inngå som inputparametre til tidsstempling når et nytt TimeStamp-Extended Tidsstempel påføres. Dette er konseptuelt i tråd med hvordan ETSI TS 101 733 håndterer Tidsstempel under ES-A profilen.

Se kap. 8.2.3 for en beskrivelse av den preprosessering som er nødvendig for å produsere det dataelementet som skal benyttes som input til tidsstempling. Når det gjelder bruk av Include elementer (ref. kap. 8.2.3) skal følgende sekvens av elementer produseres:

- ett Include element hvor URI attributtet refererer elementet SDOProfile.
- ett Include element hvor URI attributtet refererer elementet SignaturePolicyIdentifier dersom elementet er tilstede.
- ett Include element hvor URI attributtet refererer elementet SignersDocumentFormat.
- ett Include element hvor URI attributtet refererer elementet SignaturePreprosessing dersom elementet er tilstede.
- ett Include element hvor URI attributtet refererer elementet HashedData dersom elementet er tilstede.
- ett Include element hvor URI attributtet refererer elementet SigningTime dersom elementet er tilstede.
- ett Include element hvor URI attributtet refererer elementet CMSSignature.
- ett Include element hvor URI attributtet refererer elementet UserCertificateAndRevocationData dersom elementet er tilstede.
- ett Include element hvor URI attributtet refererer elementet CACertificatesAndRevocationData dersom elementet er tilstede.
- ett Include element for hvert TimeStamp-Basic element dersom disse elementene er tilstede. URI attributtet i hvert Include element vil referere ett TimeStamp-Basic element.

[11] Nøsting innebærer at et eksisterende Tidsstempel inngår som Signaturattributt ved opprettelse av et nytt Tidsstempel.

- ett Include element hvor URI attributtet refererer elementet TSCertificatesAndRevocationData dersom elementet er tilstede.
- ett Include element hvor URI attributtet refererer elementet DataValidationCertificate dersom elementet er tilstede.
- ett Include element hvor URI attributtet refererer elementet DVCCertificatesAndRevocationData dersom elementet er tilstede.
- ett Include element per allerede eksisterende TimeStamp-Extended element. URI attributtet i hvert Include element vil referere ett TimeStamp-Extended element.
- ett Include element hvor URI attributtet refererer elementet NewHashedData, som er et underelement til elementet TimeStamp-Extended. Elementet NewHashedData må derfor først populeres med en hash av Undertegnerens Dokument, dvs.:
    - en hash av verdien i SignersDocument elementet dersom dette er tilstede som et underelement av SignedObject, alternativt
    - en hash av resultatet av prosesseringen av Reference elementet i henhold til referanseprosesseringsmodellen fra XMLDSIG [8] dersom elementet er tilstede.

De samme beraktningene som er beskrevet for TimeStamp-Basic når det gjelder eventuell innlegging av Valideringsdata knyttet til tidsstemplene gjelder også for TimeStamp-Extended.

8.2.3 Preprosessering for tidsstempling

Det som skal tidsstemples er en hash påført over av ett sett av XML dataelementer. Preprosessering av dataelementene før hashing skal følge modellen fra ETSI TS 101 903 [10]. Innholdet i dette kapittelet er hentet fra nevnte dokument.

XML strukturen TimeStampType som både TimeStamp-Basic og TimeStamp-Extended inneholder en sekvens av Include elementer. Hvert Include element refererer vha. et URI attributt ett XML element som skal inngå i tidsstemplingen.

Hvert enkelt Include element skal prosesseres som følger, ref. ETSI TS 101 903 [10]:

1) Hent ut dataene som det refereres til vha. URI attributtet.
2) Dersom dataene som refereres er et ds:Reference element og referencedData attributtet er datt til verdien "true", ta resultatet av prosesseringen av det refererte ds:Reference elementet i henhold til referanseprosesseringsmodellen fra XMLDSIG [8]. I motsatt fall ta ds:Reference elementet selv.
3) Dersom dataene som refereres er et XML nodesett, skal nodesettet ”kanoniseres” (eng: canonicalize). Dersom elementet ds:Canonicalization er benyttet skal algoritmen som angis i dette elementet benyttes. I motsatt fall, skal standard metode for ”kanonisering” (eng: canonicalization) som spesifisert av XMLDSIG [8] benyttes.
4) Konkatener resulterende oktetter til de som stammer fra prosessering av tidligere Include elementer i sekvensen.

De konkatenerte oktettene benyttes til slutt som input til den nevnte hashprosessen.

9 Anbefalinger for SDO Signaturelementer av Type 2

Kapittel 8.1 beskriver hvordan mobile signaturløsninger som anvendes i dag vil ha problemer med å oppfylle de konformitetskravene som ETSI TS 101 733 stiller til Basis Signaturen.

Av den grunn er det akseptabelt at kravet i ETSI TS 101 733 om inkludering av signert sertifikatreferanse (BES/EPES krav) samt kravet om inkludering av signert referanse til Signaturpolicy (EPES krav) fravikes dersom den aktuelle signaturløsning teknisk ikke kan støtte disse, ref krav 2 i Tabell 8-2 i kap. 8.1.

Bilag A (Normativt): XML skjema

Dette bilaget beskriver XML skjemaet som er definert for SEID-SDO. Enkelte av XML elementene og XML datatypene som er benyttet i skjemaet er hentet fra XAdES [10] eller XMLDSIG [8]. En kopi av disse XML elementene/strukturene er lagt inn i kapittel B.2.

B.1 XML skjema for SEID-SDO

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:ds="http://www.w3.org/2000/09/xmldsig#" xmlns:XAdES="http://uri.etsi.org/01903/v1.2.2#"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns="http://www.npt.no/seid/xmlskjema/SDO_v1.0"
targetNamespace="http://www.npt.no/seid/xmlskjema/SDO_v1.0" elementFormDefault="qualified" attributeFormDefault="unqualified">
<xs:import namespace="http://uri.etsi.org/01903/v1.2.2#" schemaLocation="http://uri.etsi.org/01903/v1.2.2/XAdES.xsd"/>
<xs:import namespace="http://www.w3.org/2000/09/xmldsig#"
schemaLocation="http://www.w3.org/TR/2002/xmldsig-core-20020212/xmldsig-core-schema.xsd"/>

<xs:element name="SDOList">
        <xs:complexType>
                <xs:sequence maxOccurs="unbounded">
                        <xs:element ref="SDO"/>
                </xs:sequence>
        </xs:complexType>
</xs:element>

<!-- Start root element, SDO -->
<xs:element name="SDO">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="SEIDSDOVersion" type="xs:string"/>
      <xs:element ref="SDODataPart"/>
      <xs:element name="SDOSeal" minOccurs="0" maxOccurs="unbounded">
        <xs:complexType>
          <xs:choice>
            <xs:element name="SDOSignature" type="SignatureElementType"/>
            <xs:element name="SDOTimeStamp">
              <xs:complexType>
                <xs:sequence>
                        <xs:element name="TimestampValue" type="XAdES:TimeStampType"/>
                        <xs:element name="TSCertificatesAndRevocationData" type="CertificateAndRevocationType" minOccurs="0"/>
                </xs:sequence>
              </xs:complexType>
            </xs:element>
          </xs:choice>
        </xs:complexType>
      </xs:element>
      <xs:element name="Metadata" type="MetaDataType" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element name="SignedObject" type="SignedObjectType" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- End root element, SDO -->

<!-- Start SDODataPart-->
<xs:element name="SDODataPart">
        <xs:complexType>
                <xs:sequence>
                        <xs:element name="SignatureElement" type="SignatureElementType" maxOccurs="unbounded"/>
                </xs:sequence>
        </xs:complexType>
</xs:element>
<!-- end SDODataPart-->
<xs:complexType name="SignatureElementType">
        <xs:choice>
                <xs:element name="CMSSignatureElement" type="CMSSignatureElementType"/>
                <xs:element name="CAdESSignatureElement" type="XAdES:EncapsulatedPKIDataType"/>
                <xs:element name="XAdESSignatureElement" type="XMLSignatureType"/>
        </xs:choice>
</xs:complexType>
<xs:complexType name="XMLSignatureType">
```

```
            <xs:sequence>
                        <xs:element name="XAdESSignature" type="XAdESAnyType"/>
            </xs:sequence>
</xs:complexType>
<xs:complexType name="CMSSignatureElementType">
            <xs:sequence>
                        <xs:element name="SDOProfile" type="SDOProfileType"/>
                        <xs:element name="SignaturePolicyIdentifier" type="SignaturePolicyIdentifierType" minOccurs="0"/>
                        <xs:element name="SignersDocumentFormat" type="DataObjectFormatType" minOccurs="0"/>
                        <xs:element name="SignaturePreprosessing" type="SignaturePreprosessingType" minOccurs="0"/>
                        <xs:element name="HashedData" type="XAdES:DigestAlgAndValueType" minOccurs="0"/>
                        <xs:element ref="XAdES:SigningTime" minOccurs="0"/>
                        <xs:element name="CMSSignature" type="XAdES:EncapsulatedPKIDataType"/>
                        <xs:element name="UserCertificateAndRevocationData" type="CertificateAndRevocationType" minOccurs="0"/>
                        <xs:element name="CACertificatesAndRevocationData" type="CertificateAndRevocationType" minOccurs="0"/>
                        <xs:element name="TimeStamp-Basic" type="XAdES:TimeStampType" minOccurs="0" maxOccurs="unbounded"/>
                        <xs:element name="TimeStamp-Extended" type="Timestamp-ExtendedType" minOccurs="0" maxOccurs="unbounded"/>
                        <xs:element name="TSCertificatesAndRevocationData" type="CertificateAndRevocationType" minOccurs="0"/>
                        <xs:element name="DataValidationCertificate" type="XAdES:CertificateValuesType" minOccurs="0" maxOccurs="unbounded"/>
                        <xs:element name="DVCCertificatesAndRevocationData" type="CertificateAndRevocationType" minOccurs="0"/>
            </xs:sequence>
</xs:complexType>
<xs:complexType name="DataObjectFormatType">
            <xs:sequence>
                        <xs:element name="Description" type="xs:string" minOccurs="0"/>
                        <xs:element name="ObjectIdentifier" type="XAdES:ObjectIdentifierType" minOccurs="0"/>
                        <xs:element name="MimeType" type="xs:string" minOccurs="0"/>
                        <xs:element name="Encoding" type="xs:anyURI" minOccurs="0"/>
            </xs:sequence>
</xs:complexType>
<xs:complexType name="Timestamp-ExtendedType">
            <xs:sequence>
                        <xs:element name="NewHashedData" type="XAdES:DigestAlgAndValueType" minOccurs="0"/>
                        <xs:element name="TimestampValue" type="XAdES:TimeStampType"/>
            </xs:sequence>
</xs:complexType>
<xs:simpleType name="SignaturePreprosessingType">
            <xs:restriction base="xs:string">
                        <xs:enumeration value="Hashed"/>
                        <xs:enumeration value="HashedAndTimeAppended"/>
            </xs:restriction>
</xs:simpleType>
<xs:complexType name="SignaturePolicyIdentifierType">
            <xs:choice>
                        <xs:element name="SignaturePolicyId" type="SignaturePolicyIdType"/>
                        <xs:element name="SignaturePolicyImplied"/>
            </xs:choice>
</xs:complexType>
<xs:complexType name="SignaturePolicyIdType">
            <xs:sequence>
                        <xs:element name="SigPolicyId" type="XAdES:ObjectIdentifierType"/>
                        <xs:element ref="ds:Transforms" minOccurs="0"/>
                        <xs:element name="SigPolicyHash" type="XAdES:DigestAlgAndValueType" minOccurs="0"/>
                        <xs:element name="SigPolicyQualifiers" type="XAdES:SigPolicyQualifiersListType" minOccurs="0"/>
            </xs:sequence>
</xs:complexType>
<xs:complexType name="CertificateAndRevocationType">
            <xs:sequence>
                        <xs:element name="CertificateValues" type="XAdES:CertificateValuesType" minOccurs="0"/>
                                    <xs:element name="RevocationValues" type="XAdES:RevocationValuesType" minOccurs="0"/>
                        <xs:element name="CRLrefs" type="XAdES:CRLRefsType" minOccurs="0" maxOccurs="unbounded"/>
            </xs:sequence>
</xs:complexType>

<xs:complexType name="MetaDataType">
   <xs:sequence>
      <xs:element name="Include" type="XAdES:IncludeType" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element name="ValuePair" minOccurs="0" maxOccurs="unbounded">
         <xs:complexType>
            <xs:sequence>
               <xs:choice>
                  <xs:element name="Code" minOccurs="0">
                     <xs:complexType>
                        <xs:sequence>
                           <xs:element name="CodeListId">
                              <xs:complexType>
```

```
                                        <xs:sequence>
                                                <xs:element name="Organisation" type="xs:string" minOccurs="0"/>
                                                <xs:element name="CodeListIdentifier" type="xs:string" minOccurs="0"/>
                                                <xs:element name="Version" type="xs:string" minOccurs="0"/>
                                        </xs:sequence>
                                    </xs:complexType>
                                </xs:element>
                                <xs:element name="CodeId" type="xs:string"/>
                            </xs:sequence>
                        </xs:complexType>
                    </xs:element>
                    <xs:element name="Name" type="xs:string" minOccurs="0"/>
                </xs:choice>
                <xs:element name="Description" type="xs:string" minOccurs="0"/>
                <xs:element name="Value" type="xs:string" minOccurs="0"/>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="SignedObjectType">
        <xs:choice>
                <xs:element name="SignersDocument" type="xs:base64Binary" minOccurs="0"/>
                <xs:element ref="ds:Reference" minOccurs="0"/>
        </xs:choice>
</xs:complexType>

<xs:simpleType name="SDOProfileType">
        <xs:restriction base="xs:string">
                <xs:enumeration value="SEID-SDO-Basic"/>
                <xs:enumeration value="SEID-SDO-Basic-V"/>
                <xs:enumeration value="SEID-SDO-Basic-T"/>
                <xs:enumeration value="SEID-SDO-Basic-TSP"/>
                <xs:enumeration value="SEID-SDO-Basic-Extended"/>
                <xs:enumeration value="SEID-SDO-Basic-Archive"/>
        </xs:restriction>
</xs:simpleType>

<!-- Start SignatureAnyType -->
<xs:complexType name="XAdESAnyType" mixed="true">
        <xs:sequence>
                <xs:any namespace="http://uri.etsi.org/01903/v1.2.2#" processContents="lax"/>
                <xs:any namespace="http://www.w3.org/2000/09/xmldsig#" processContents="lax"/>
        </xs:sequence>
        <xs:anyAttribute namespace="##any"/>
</xs:complexType>
<!-- End AnyType -->

</xs:schema>
```

B.2 XML datastrukturer importert fra XAdES [10] og XMLDSIG [8]

```
<!-- Start EncapsulatedPKIDataType-->
<xsd:complexType name="EncapsulatedPKIDataType">
        <xsd:simpleContent>
                <xsd:extension base="xsd:base64Binary">
                        <xsd:attribute name="Id" type="xsd:ID" use="optional"/>
                </xsd:extension>
        </xsd:simpleContent>
</xsd:complexType>
<!-- End EncapsulatedPKIDataType -->

<!-- Start TimeStampType -->
<xsd:complexType name="TimeStampType">
        <xsd:sequence>
                <xsd:element name="Include" type="IncludeType" maxOccurs="unbounded"/>
                <xsd:element ref="ds:CanonicalizationMethod" minOccurs="0"/>
                <xsd:choice>
                        <xsd:element name="EncapsulatedTimeStamp" type="EncapsulatedPKIDataType"/>
                        <xsd:element name="XMLTimeStamp" type="AnyType"/>
                </xsd:choice>
        </xsd:sequence>
        <xsd:attribute name="Id" type="xsd:ID" use="optional"/>
```

```
</xsd:complexType>
<!-- End TimeStampType -->

<!-- Start IncludeType-->
<xsd:complexType name="IncludeType">
        <xsd:attribute name="URI" type="xsd:anyURI" use="required"/>
        <xsd:attribute name="referencedData" type="xsd:boolean" use="optional"/>
</xsd:complexType>
<!-- End IncludeType-->

<!-- Start DigestAlgAndValueType -->
<xsd:complexType name="DigestAlgAndValueType">
        <xsd:sequence>
                <xsd:element ref="ds:DigestMethod"/>
                <xsd:element ref="ds:DigestValue"/>
        </xsd:sequence>
</xsd:complexType>
<!-- End DigestAlgAndValueType -->

<xsd:element name="SigningTime" type="xsd:dateTime" minOccurs="0"/>

<!-- Start CertificateValuesType -->
<xsd:complexType name="CertificateValuesType">
        <xsd:choice minOccurs="0" maxOccurs="unbounded">
                <xsd:element name="EncapsulatedX509Certificate" type="EncapsulatedPKIDataType"/>
                <xsd:element name="OtherCertificate" type="AnyType"/>
        </xsd:choice>
        <xsd:attribute name="Id" type="xsd:ID" use="optional"/>
</xsd:complexType>
<!-- End CertificateValuesType -->


<!-- Start ObjectIdentifierType-->
<xsd:complexType name="ObjectIdentifierType">
        <xsd:sequence>
                <xsd:element name="Identifier" type="IdentifierType"/>
                <xsd:element name="Description" type="xsd:string" minOccurs="0"/>
                <xsd:element name="DocumentationReferences" type="DocumentationReferencesType" minOccurs="0"/>
        </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="IdentifierType">
        <xsd:simpleContent>
                <xsd:extension base="xsd:anyURI">
                        <xsd:attribute name="Qualifier" type="QualifierType" use="optional"/>
                </xsd:extension>
        </xsd:simpleContent>
</xsd:complexType>
<xsd:simpleType name="QualifierType">
        <xsd:restriction base="xsd:string">
                <xsd:enumeration value="OIDAsURI"/>
                <xsd:enumeration value="OIDAsURN"/>
        </xsd:restriction>
</xsd:simpleType>
<xsd:complexType name="DocumentationReferencesType">
        <xsd:sequence maxOccurs="unbounded">
                <xsd:element name="DocumentationReference" type="xsd:anyURI"/>
        </xsd:sequence>
</xsd:complexType>
<!-- End ObjectIdentifierType-->

<!-- Start SigPolicyQualifiersType -->
<xsd:complexType name="SigPolicyQualifiersListType">
        <xsd:sequence>
                <xsd:element name="SigPolicyQualifier" type="AnyType" maxOccurs="unbounded"/>
        </xsd:sequence>
</xsd:complexType>
<!-- end SigPolicyQualifiersType -->

<!-- Start CertificateValuesType -->
<xsd:complexType name="CertificateValuesType">
        <xsd:choice minOccurs="0" maxOccurs="unbounded">
                <xsd:element name="EncapsulatedX509Certificate" type="EncapsulatedPKIDataType"/>
                <xsd:element name="OtherCertificate" type="AnyType"/>
        </xsd:choice>
        <xsd:attribute name="Id" type="xsd:ID" use="optional"/>
</xsd:complexType>
```

```
<!-- End CertificateValuesType -->


<!-- Start RevocationValuesType -->
<xsd:complexType name="RevocationValuesType">
	<xsd:sequence>
		<xsd:element name="CRLValues" type="CRLValuesType" minOccurs="0"/>
		<xsd:element name="OCSPValues" type="OCSPValuesType" minOccurs="0"/>
		<xsd:element name="OtherValues" type="OtherCertStatusValuesType" minOccurs="0"/>
	</xsd:sequence>
	<xsd:attribute name="Id" type="xsd:ID" use="optional"/>
</xsd:complexType>
<xsd:complexType name="CRLValuesType">
	<xsd:sequence>
		<xsd:element name="EncapsulatedCRLValue" type="EncapsulatedPKIDataType" maxOccurs="unbounded"/>
	</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="OCSPValuesType">
	<xsd:sequence>
		<xsd:element name="EncapsulatedOCSPValue" type="EncapsulatedPKIDataType" maxOccurs="unbounded"/>
	</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="OtherCertStatusValuesType">
	<xsd:sequence>
		<xsd:element name="OtherValue" type="AnyType" maxOccurs="unbounded"/>
	</xsd:sequence>
</xsd:complexType>
<!-- End RevocationValuesType -->

<!—Start CRLRefsType -->
<xsd:complexType name="CRLRefsType">
	<xsd:sequence>
		<xsd:element name="CRLRef" type="CRLRefType" maxOccurs="unbounded"/>
	</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="CRLRefType">
	<xsd:sequence>
		<xsd:element name="DigestAlgAndValue" type="DigestAlgAndValueType"/>
		<xsd:element name="CRLIdentifier" type="CRLIdentifierType" minOccurs="0"/>
	</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="CRLIdentifierType">
	<xsd:sequence>
		<xsd:element name="Issuer" type="xsd:string"/>
		<xsd:element name="IssueTime" type="xsd:dateTime"/>
		<xsd:element name="Number" type="xsd:integer" minOccurs="0"/>
	</xsd:sequence>
	<xsd:attribute name="URI" type="xsd:anyURI" use="optional"/>
</xsd:complexType>
<!-- End CRLRefsType -->

<!-- Start AnyType -->
<xsd:complexType name="AnyType" mixed="true">
	<xsd:sequence minOccurs="0" maxOccurs="unbounded">
		<xsd:any namespace="##any" processContents="lax"/>
	</xsd:sequence>
	<xsd:anyAttribute namespace="##any"/>
</xsd:complexType>
<!-- End AnyType -->

<!-- Start SignaturePolicyIdentifierType-->
<!-- Vi har laget en ny variant av denne typen hvor elementet SigPolicyHash er gjort opsjonelt.-->
<xsd:complexType name="SignaturePolicyIdentifierType">
	<xsd:choice>
		<xsd:element name="SignaturePolicyId" type="SignaturePolicyIdType"/>
		<xsd:element name="SignaturePolicyImplied"/>
	</xsd:choice>
</xsd:complexType>
<xsd:complexType name="SignaturePolicyIdType">
	<xsd:sequence>
		<xsd:element name="SigPolicyId" type="ObjectIdentifierType"/>
		<xsd:element ref="ds:Transforms" minOccurs="0"/>
		<xsd:element name="SigPolicyHash" type="DigestAlgAndValueType"/>
		<xsd:element name="SigPolicyQualifiers" type="SigPolicyQualifiersListType" minOccurs="0"/>
	</xsd:sequence>
</xsd:complexType>
```

```
<xsd:complexType name="SigPolicyQualifiersListType">
        <xsd:sequence>
                <xsd:element name="SigPolicyQualifier" type="AnyType" maxOccurs="unbounded"/>
        </xsd:sequence>
</xsd:complexType>
<!-- End SignaturePolicyIdentifierType -->

<!-- Start DataObjectFormatType -->
<!-- Vi har laget en ny variant av denne typen hvor attributtet Objectreference er fjernet -->
<xsd:complexType name="DataObjectFormatType">
        <xsd:sequence>
                <xsd:element name="Description" type="xsd:string" minOccurs="0"/>
                <xsd:element name="ObjectIdentifier" type="ObjectIdentifierType" minOccurs="0"/>
                <xsd:element name="MimeType" type="xsd:string" minOccurs="0"/>
                <xsd:element name="Encoding" type="xsd:anyURI" minOccurs="0"/>
        </xsd:sequence>
        <xsd:attribute name="ObjectReference" type="xsd:anyURI" use="required"/>
</xsd:complexType>
<!-- End DataObjectFormatType -->
```

Bilag B (Normativt): XML elementbeskrivelse

Tabell B-1 beskriver alle elementene i XML skjemaet i Bilag A som er definert av SEID-prosjektet. I de tilfeller hvor det er hentet XML elementer eller XML datatyper fra XMLDIG [8] eller XAdES [10] så henvises det til disse dokumentene for nærmere beskrivelse av hvordan disse elementene/strukturene skal brukes.

#	XML element	Beskrivelse av innhold	Datatype og format
1	**SDOList**	Inneholder en liste med SDO objekter. Vanligvis vil denne listen kun inneholde ett element, men her gis muligheter til å gruppere flere SDOer sammen.	*Sekvens* av **SDO** elementer
2	**SDO**	Inneholder en eller flere signaturer og all Valideringsdata knyttet til disse.	*Sekvens* av elementene {**SEIDSDOVersion, SDODataPart, SDOSeal, Metadata, SignedObject**}
2-1	**SEIDSDOVersion**	Angir versjon av XML skjemaet i Bilag A. Skal være satt til 1.0.	DataType: Streng Format: <nummer>.<nummer>
2-2	**SDODataPart**	Den delen av SEID-SDO som inneholder SDO Signaturelementer (se kap. 6.4).	*Sekvens* bestående av ett eller flere elementer av typen **SignatureElement**
2-3	**SDOSeal**	Signatur eller tidsstempel generert over SDODataPart.	*Forekomst* av **SDOSignature** element ELLER *Forekomst* av **SDOTimestamp** element
2-3-1	**SDOSignature**	For lagring av signatur over SDODataPart.	DataType: **SignatureElementType** Tilsvarer strukturen beskrevet for elementet **SignatureElement** i pkt.3.
2-3-2	**SDOTimestamp**	Struktur for lagring av ett tidsstempel over SDODataPart og tilhørende Valideringsdata.	*Sekvens* av elementene {**TimestampValue, TSCertificatesAndValidationdata**}
2-3-2-1	**TimestampValue**	For lagring av tidsstempel over SDODataPart.	DataType: **TimestampType** ihht. XAdES
2-3-2-2	**TSCertificates AndValidationdata**	For lagring av Valideringsdata knyttet til tidsstempelets signatur.	DataType: **CertificateAndRevocationType** Tilsvarer strukturen beskrevet for elementet **UserCertificateAndRevocationdata** i pkt.3-1-8.
2-4	**Metadata**	Tilleggsinformasjon om SDO objektet som helhet og/eller dets enkelte elementer. Kan brukes for legge på for eksempel Dublin Core metadata[12] , tilstand eller kontekstinformasjon relevant for et SDO. Metadata som legges inn bør være forståelig for fremtidige tolkere av SDOet.	*Sekvens* av elementene {**CodeListId, Include, ValuePair**}
2-4-1	**Include**	Peker(e) til de elementer i SEID-SDO som metadataene refererer til.	DataType: **IncludeType** fra XAdES Se XAdES for detaljer.
2-4-2	**Valuepair**	Struktur for lagring av selve metadataene.	*Sekvens* av elementene {**Code** *ELLER* **Name, Description, Value**}
2-4-2-1	**Code**	Struktur for å angi en metadatakode som benyttes.	Sekvens av {CodeListId, CodeId}
2-4-2-2	**Name**	Angir navnet på en kode, hvis koden ikke er definert i noen standard. Bør kun brukes sammen med en god generell beskrivelse som bør være kandidat for standardisering.	DataType: Streng Format: Ingen
2-4-2-1-1	**CodeListId**	Identifiserer en bestemt versjon av en kodeliste og hvem som administrerer denne.	*Sekvens* av elementene {**Organisation, CodeListIdentifier, Version**}

[12] http://dublincore.org/

#	XML element	Beskrivelse av innhold	Datatype og format
2-4-2-1-1-1	**Organisation**	Navnet på den enhet (eksempel ISO standard xxx, Dublin Core, eller Post og Teletilsynet) som har ansvaret for å utarbeide og vedlikeholde en liste over definerte koder.	DataType: Streng Format: Ingen
2-4-2-1-1-2	**CodeListIdentifier**	En ID som identifiserer entydig globalt eller inne organisasjonen hvilken kodeliste som benyttes	DataType: Streng Format: Ingen
2-4-2-1-1-3	**Version**	Hvilken versjon av kodelisten som benyttes.	DataType: Streng Format: Ingen
2-4-2-1-2	**CodeID**	En kode som gir mening enten alene eller sammen med et utfylt Value-element. Eksempel på CodeID brukt uten Value kan være format converted.Benyttet for å angi at et originaldokumentet i et SDO er formatkonvertert.	DataType: Streng Format: Ingen
2-4-2-3	**Description**	Tekstlig beskrivelse av verdi/kode metadataene. For koden dc:Rights Management[13], kunne en lagt inn forklaringen: ”Information about rights held in and over the resource”.	DataType: Streng Format: Ingen
2-4-2-4	**Value**	For lagring av selve metadataverdien.	DataType: Streng Format: Ingen
2-5	**SignedObject**	Inneholder Undertegnerens Dokument, evt. en referanse til dette.	*Forekomst* av elementet **SignersDocument** ELLER *Forekomst* av elementet **Reference**
2-5-1	**SignersDocument**	For lagring av Undertegnerens Dokument.	DataType: **AnyType** fra XAdES
2-5-2	**Reference**	Referanse til Undertegnerens Dokument.	DataType: **Reference** fra XMLDSIG Se XMLDSIG for detaljer.
3	**SignatureElement**	Inneholder ett av de tre typene SDO Signaturelementer som er definert i kap. 6.4	*Forekomst* av ett av følgende tre elementer: {**CMSSignatureElement, CAdESSignatureElement, XAdESSignatureElement}**
3-1	**CMSSignature Element**	Inneholder en enkelt CMS signatur og tilhørende Valideringsdata knyttet til denne.	Sekvens av følgende elementer: {**SDOProfile, SignaturePolicyIdentifier, SignersDocumentFormat, SignaturePreprosesseing, HashedData, SigningTime, CMSSignature, UserCertificateAndRevocationData, CACertificatesAndRevocationData, TimeStamp-Basic, TimeStamp-Extended, TSCertificatesAndRevocationData, DataValidationCertificate, DVCCertificatesAndRevocationData**}
3-1-1	**SDOProfile**	Angir profil ihht. kap. 7.1.1.	DataType: Streng Format: Må ha en av følgende seks verdier **{SEID-SDO-Basic, SEID-SDO-Basic-V, SEID-SDO-Basic-T, SEID-SDO-TSP, SEID-SDO-Extended, SEID-SDO-Archive}**
3-1-2	**SignaturePolicyIdentifier**	En entydig OID som angir hvilken signaturpolicy som er benyttet ved signering, evt. for angivelse av at signaturpolicy anses implisitt kjent ut ifra kontekst.	DataType: **SignaturePolicyIdentifierType** Denne typen er identisk med tilsvarende type fra XAdES med unntak av at elementet SigpolicyHash er gjort opsjonelt. Se XAdES for detaljer.

[13] http://purl.org/dc/elements/1.1/rights

#	XML element	Beskrivelse av innhold	Datatype og format
3-1-3	**SignersDocumentFormat**	Angir dataformat for Undertegnerens Dokument.	DataType: **DataObjectFormatType** Denne typen er identisk med tilsvarende type fra XAdES med unntak av at attributtet Objectreference er fjernet. Se XAdES for detaljer.
3-1.4	**SignaturePreprosessing**	Angir hva slags preprosessering som er foretatt på Undertegnerens Dokument i forkant av CMS signering, se kap. 8.2.3.	DataType: Streng Format: Må ha en av følgende to verdier {Hashed, HashedAndTimeAppended}
3-1-5	**HashedData**	Hash av Undertegnerens Dokument	DataType: **DigestAlgAndValueType** fra XAdES Se XAdES for detaljer.
3-1-6	**SigningTime**	Tidsangivelse for når signaturen ble opprettet. Tidsangivelsen er en påstand knyttet til Undertegneren og dennes signaturfremstillingssystem[14].	DataType: **SigningTime** fra XAdES Se XAdES for detaljer.
3-1-7	**CMSSignature**	For lagring av CMS Signaturobjektet.	DataType: **EncapsulatedPKIDataType** fra XAdES Se XAdES for detaljer.
3-1-8	**UserCertificateAnd RevocationData**	For lagring av Undertegnerens Signatursertifikat samt Valideringsdata (CRL referanse eller OCSP respons) knyttet til dette sertifikatet.	*Sekvens* av elementene {**CertificateValue**, **RevocationValues, CRLrefs**}
3-1-8-1	**CertificateValue**	For lagring av Undertegnerens sertifikat.	DataType: **CertificateValuesType** fra XAdES Se XAdES for detaljer.
3-1-8-2	**RevocationValues**	For lagring av aktuell CRL og/eller OCSP respons.	DataType: **RevocationValuesType** fra XAdES Se XAdES for detaljer.
3-1-8-3	**CRLrefs**	For lagring av referanser til aktuell CRL.	DataType: **CRLRefsType** fra XAdES Se XAdES for detaljer.
3-1-9	**CACertificatesAnd Revocationdata**	For lagring av CA sertifikater og Valideringsdata for hele CA sertifikatkjeden knyttet til CMS signaturen.	Tilsvarende struktur som i 3-1-8
3-1-10	**TimeStamp-Basic**	Struktur for lagring av et tidsstempel ihht. kap. 8.2.1.	DataType: **TimeStampType** fra XAdES Se XAdES for detaljer.
3-1-11	**TimeStamp-Extended**	Struktur for lagring av et tidsstempel ihht. kap. 8.2.2.	*Sekvens* av elementene {**NewHashedData**,**TimeStampValue**}
3-1-11-1	**NewHashedData**	Inneholder hash av Undertegnerens Dokument.	DataType: **DigestAlgAndValueType** fra XAdES Se XAdES for detaljer.
3-1-11-2	**TimestampValue**	For lagring av tidsstempel generert ihht. kap. 8.2.2.	DataType: **TimeStampType** fra XAdES Se XAdES for detaljer.
3-1-12	**TSCertificates AndRevccationData**	For lagring av CA sertifikater og Valideringsdata for CA sertifikatkjeden relatert til alle tidsstemplene av typen Timestamp-Basic og TimeStamp-Extended.	Tilsvarende struktur som i 3-1-8 og 3-1-9
3-1-13	**DataValidation Certificate**	Et Datavalideringssertifkat som bekrefter gyldigheten av CMS signaturen.	DataType: **CertificateValuesType** fra XAdES Se XAdES or detaljer.
3-1-14	**DVCCertificatesAnd RevocationData**	For lagring av CA sertifikater og Valideringsdata for CA sertifikatkjeden relatert til Datavalideringssertifikatet i 3-1-13.	Tilsvarende struktur som i 3-1-8, 3-1-9 og 3-1-12
3-2	**CAdESSignatureElement**	For lagring av et CAdES konform CMS signatur.	DataType: **EncapsulatedPKIDataType** fra XAdES Se XAdES or detaljer.
3-3	**XAdESSignatureElement**	Struktur for lagring av en XAdES konform XMLDSIG signatur.	*Sekvens* bestående av elementet **XAdESSignature**.
3-3-1	**XAdESSignature**	for lagring av en XAdES konform XMLDSIG signatur.	DataType: **XADESAnyType**

[14] En slik tid har naturligvis forskjellig verdi avhengig av hvem som signerer.

Tabell B-1 – Beskrivelse av XML elementer definert for SDO Signaturelementer av Type 1

Bilag C (Informativt): Standardisering av signaturformater i ETSI

C.1 Introduksjon

ETSI har i lengre tid arbeidet med å definere formater for elektroniske signaturer som skal ha gyldighet over tid. Resultatet av arbeidet er publisert i to tekniske standarder:

- ETSI TS 101 733 [9] som fokuserer på CMS [7] formaterte signaturer.
- ETSI TS 101 903 (XAdES) [10] som fokuserer på XMLDSIG [8] formaterte signaturer.

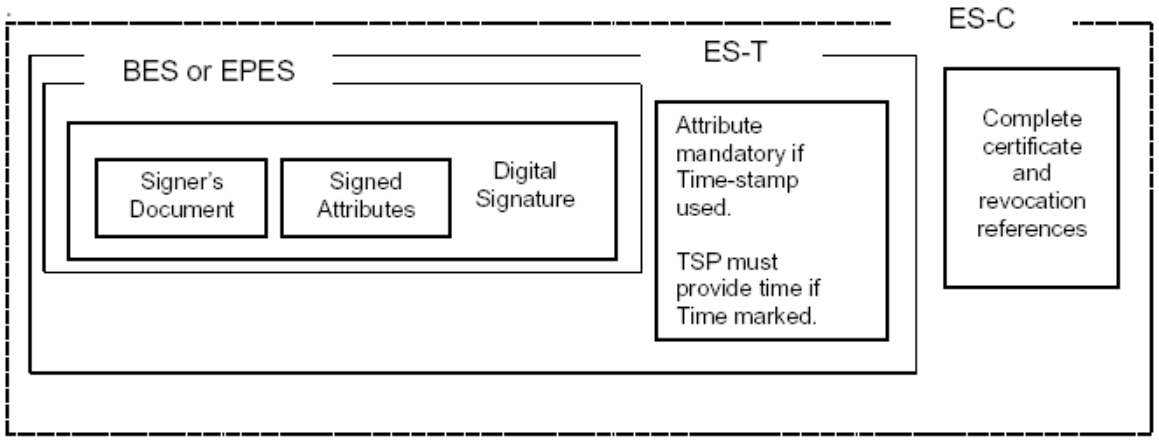
De nevnte dokumentene har også fått tilsvarende publikasjoner i IETF:

- RFC 3126 [11] tilsvarer ETSI TS 101 733 v1.2.2, dvs. en forgjenger til versjon 1.5.1 [9].
- RFC 3275 [8] tilsvarer ETSI TS 101 903 v.1.1.1, dvs. en forgjenger til versjon 1.2.2 [10].

Spesifikasjonene er basert på, og således harmonisert med, eksisterende internasjonale standarder for elektroniske signaturformater [7] [8].

Kort oppsummert så definerer ETSI TS 101 733 signerte og usignerte ASN.1 attributter som kan legges til en basis CMS signatur [7] og representerer således en utvidelse av CMS standarden. XADES definerer tilsvarende XML utvidelser for en basis XMLDSIG [8] signatur.

C.2 Formatprofiler definert av ETSI



Figur C1 – Eksempler på formatprofiler som definert av ETSI

Basert på de generelle signaturformatene som ETSI standardene bygger på har ETSI definert ulike formatprofiler som skal dekke ulike behov for langtidslagring.
ETSI definerer to typer basissignaturer.

- **BES (Basic Electronic Signature)** er et minimumsformat for en elektronisk signatur. Den inneholder et dataelement (dvs. de data som i utgangspunktet skal signeres),

eventuelle tilleggsattributter samt selve signaturen over disse. Eventuelle usignerte attributter kan inkluderes som en opsjon.

- **EPES (Explicit Policy-based Electronic Signature)** bygger på BES men inneholder i tillegg en Signaturpolicy identifikator som et obligatorisk signert attributt. EPES angir med andre ord eksplisitt den aktuelle Signaturpolicy, i motsetning til BES, hvor den aktuelle Signaturpolicy implisitt kan angis av den aktuelle brukskontekst eller semantikken til de signerte data.

Med utgangspunkt i disse to basissignaturene definer ETSI ytterligere og mer omfattende formatprofiler hvor langtidslagring er fokus. Disse profilene innebærer i praksis at signaturformatet utvides til å støtte innlegging av ulike typer Valideringsdata[15] i form av usignerte tilleggsattributter.

Følgende mer omfattende formatprofiler er definert av ETSI:

- **ES-T (ES with Time Stamp):** Med utgangspunkt i BES eller EPES, legges det til tiltrodd tid i form av Tidsstempel over signaturen. Alternativt kan en tiltrodd tidsangivelse (såkalt tidsmerke) for signaturen vedlikeholdes av en ekstern part.
- **ES-C (ES with Complete validation reference data):** Med utgangspunkt i ES-T, legges det til komplette referanser til aktuelle CA-sertifikater i tillitskjeden for den aktuelle signatur samt tilhørende referanse til nødvendig sertifikatstatusinformasjon (CRLer, OCSP responser).
- **ES-X (ES with eXtended validation data):** med utgangspunkt i ES-C, legges det til enten et Tidsstempel (over hele ES-C eller kun referanseinformasjonen nevnt i forrige punkt) og/eller de faktisk Valideringsdata som ES-C refererer til. ETSI definerer 4 typer av ES-X (ES-X Long, ES-X Type 1, ES-X Type 2, ES-X Long Type 1 og 2). Leseren henvises til [9] for eventuelle detaljer vedrørende disse.
- **ES-A (ES with Archiving validation data):** med utgangspunkt i ES-X, legges det til ett eller flere Tidsstempel over ES-X.

Figur C1 er hentet fra ETSI TS 101 733 og illustrerer på en god måte hvordan de ulike profilene bygger på hverandre. Formatprofilene ES-X og ES-A er ikke vist i figuren. Leseren henvises til ETSI standarden for detaljer vedrørende disse.

Krav til konformitet mot ETSI TS 101 733 er definert for alle formatprofiler med unntak av ES-X og ES-A. Det er med andre ord ikke nødvendig å implementere støtte for de to sistnevnte formatprofilene for å være konform med standarden.

[15] Eksempler på Valideringsdata er sertifikater, sertifikatstatus for de samme sertifikatene samt Tidsstempel.

Bilag D (Informativt): Elektroniske signaturer og langtidslagring

D.1 Innledning

Dette Bilaget redegjør for sentrale utfordringer knyttet til digitalt signerte meldinger som skal ha lang levetid. Fokus er viet de utfordringene som kan løses ved bruk av SEID-SDO.

D.2 Utfordringer som kan løses ved utvidet meldingsformat

Utfordring A) Begrenset levetid på signaturnøkkel

Dersom uautoriserte har fått tilgang til et sertifikats private nøkkel kan de benytte disse til å lage falske signaturer. *Selv om et sertifikat er revokert så kan uautoriserte signere et dokument og påstå at det ble signert før sertifikatet var revokert.* Utfordringen her ligger i å kunne bevise om en digital signatur eksisterte før eller etter at sertifikatet ble revokert. Videre er det en utfordring at den kryptografiske styrken til en signeringsnøkkel en dag vil gå ut på dato.

Løsning: SEID-SDO-Basic-T

SEID-SDO-Basic-T benytter et tiltrodd Tidsstempel over den opprinnelige signaturen. Et Tidsstempel er mer enn bare en tidsangivelse. Et Tidsstempel inneholder en signatur over den opprinnelige signaturen. Således fungerer den tiltrodde tidsstemplingstjenesten som en garantist for at den opprinnelige signaturen eksisterte før et gitt tidspunkt.

SEID-SDO-Basic-T gjør at man kan bevise:

- at signaturen ble utført før sertifikatet ble tilbakekalt.
- at signaturen ble utført på et tidspunkt da nøkkelen fortsatt var kryptografisk sterk nok.

Utfordring B) Arkivering av sertifikater og tilbakekallingsinformasjon

En stor utfordring knyttet til langtidslagring av signerte elektroniske dokumenter er å ta vare på all den informasjonen som er nødvendig for å kunne validere signaturen. Man må blant annet ta vare på sertifikatene til de tiltrodde Sertifikatutstederne, de tiltrodde sertifikatstatustjenestene og de tiltrodde tidsstemplingstjenestene. Videre må man ta vare på sertifikatstatusinformasjonen, enten det er i form av OCSP responser eller i form av CRL lister.

Løsning: SEID-SDO-Extended

Meldingsformatet SEID-SDO-Extended løser dette problemet ved at sertifikater og Valideringsdata lagres sammen med meldingen.

SEID-SDO-Extended legger til rette for at man kan bevise:

- at sertifikatet er utstedt av en tiltrodd utsteder, fordi man har alle nødvendige sertifikater for å bygge tillitskjeden.

- at sertifikatet ikke var revokert på tidsstemplingstidspunktet (merk: ikke signeringstidspunktet), fordi man har nødvendige CRL eller OCSP responser.

Utfordring C) Begrenset levetid på signatur hash, Tidsstempel og Valideringsdata

Signeringsalgoritmer og hash algoritmer har en begrenset levetid. Siden sertifikater og sertifikatstatusinformasjon også er signerte objekter så har de tilsvarende begrenset levetid som de signerte dataobjektene (BES og EPES). Videre kan man også komme i en situasjon hvor en tidsstemplingstjeneste blir kompromittert.

Løsning: SEID-SDO-Archive

Formatet SEID-SDO-Archive signerer det aller meste av innholdet i meldingene. Formatet kan benyttes rekursivt for å beskytte meldingene ettersom nøkler blir svakere.

SEID-SDO-Archive gjør at man kan bevise:

- at den opprinnelige hash algoritmen var kryptografisk sterk nok på den tiden da dokumentet ble tidsstemplet (eks. ved at det blir gjort en SHA256 også over de *opprinnelige data* som fra før bare er signert med SHA1).
- at Tidsstempelet fra SEID-SDO-Basic-T eksisterte før evt. tidsstemplingssertifikater blir kompromittert.
- at tidligere Tidsstempel (TimeStamp-Basic og TimeStamp-Extended) ble utført på et tidspunkt da disse var kryptografisk sterke nok.

Bilag E (Informativt): SEID-SDO og Kvalifiserte Signaturer

SEID-SDO skal støtte lagring og arkivering av Elektroniske Signaturer generelt, herunder Kvalifiserte Signaturer[16]. En Kvalifisert Signatur er en Elektronisk Signatur som er knyttet til et kvalifisert sertifikat og fremstilt av et godkjent sikkert signaturfremstillingssystem (se lov om elektronisk signatur [16]).

Reglene om kvalifiserte sertifikater[17] stiller bestemte krav til sertifikatet og til Sertifikatutsteder. Dette dokumentet her stiller ikke slike krav for noen av sertifikatene som er benyttet i SEID-SDO.

Reglene om Kvalifiserte Signaturer stiller bestemte krav til signaturfremstillingssystemet. Dette dokumentet stiller ikke slike krav til noen av signaturene som inngår i SEID-SDO.

Innpakning av en brukers Elektroniske Signatur i et SEID-SDO betyr ikke at brukersignaturen blir hverken mer eller mindre kvalifisert enn det signaturen var før innpakning.

Innpakning av en brukers Elektroniske Signatur i et SEID-SDO knytter signaturen sammen med attributter som sikrer signaturen gyldighet over tid, for eksempel slik at det på tidspunkt etter at de kryptografiske nøklene er kompromittert fremdeles kan bevises at signaturen var gyldig på et tidspunkt før de kryptografiske nøklene ble kompromittert.

Verdien av dette beviset er naturligvis avhengig av den sikkerhetsmessige styrken på de signaturer som inngår i SEID-SDO, herunder signaturer benyttet ved eventuell tidsstempling.

[16] Jf. Lov om Elektronisk Signatur [16], §3 nr.3.
[17] Jf. Lov om Elektronisk Signatur [16], §4.